

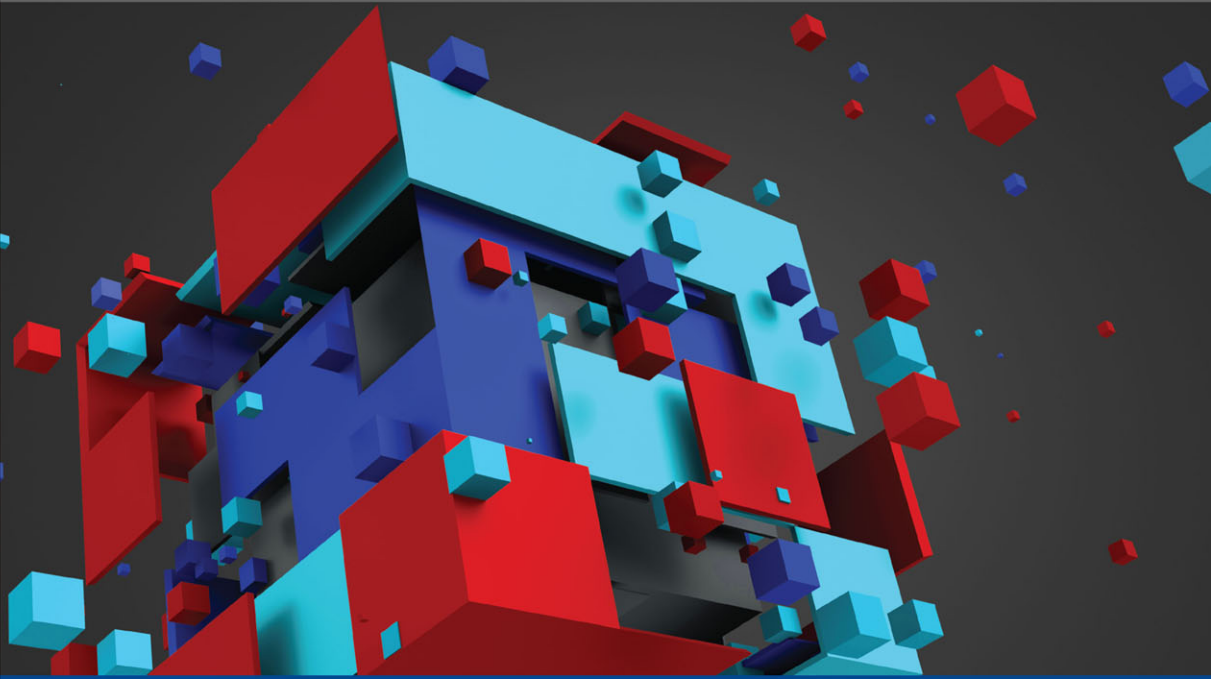
"This book is an essential guide for every architect and developer deploying secure, business-critical solutions on Azure."

—from the foreword by **Scott Guthrie**,
Executive Vice President, Cloud + AI Group, Microsoft



Designing and Developing Secure Azure Solutions

Best practices



Michael Howard • Simone Curzi • Heinrich Gantenbein

Designing and Developing Secure Azure Solutions

Michael Howard
Heinrich Gantenbein
Simone Curzi

Designing and Developing Secure Azure Solutions

Table of Contents

Cover

Title Page

Copyright Page

Contents at a glance

Contents

Acknowledgments

Foreword

About the Authors

Introduction

PART I: SECURITY PRINCIPLES

Chapter 1 Secure development lifecycle processes

Developers are the number-one source of compromises

Introducing the Microsoft Security Development Lifecycle

Quality = security

Securing features vs. security features

SDL components

Security training

Defining your bug bar

Attack surface analysis

Threat modeling

Defining your toolchain

Avoiding banned functionality

Using static analysis tools

Table of Contents

- Using dynamic analysis tools
- Security code review
- Having an incident response plan
- Performing penetration tests

SDL tasks by sprint

The human element

Summary

Chapter 2 Secure design

The cloud, DevOps, and security

IaaS vs. PaaS vs. SaaS, and the shared responsibility

Zero trust for developers

Thinking about secure design

Security design principles applied to Azure

- Attack surface reduction
- Complete mediation
- Defense in depth
- Economy of mechanisms
- Fail-safe defaults
- Fail safe and fail secure
- Least common mechanism
- Least privilege
- Leveraging existing components
- Open design
- Psychological acceptability
- Separation of duties
- Single point of failure
- Weakest link

Summary

Chapter 3 Security patterns

What is a pattern?

Our take on Azure security patterns

Table of Contents

Authentication pattern

- Use a centralized identity provider for authentication

Authorization patterns

- Adopt just-in-time administration

- Assign roles to groups

- Isolate from the internet

- Isolate with an identity perimeter

- Use role-based access control (RBAC)

Secrets management patterns

- Use managed identities

- Protect secrets with Azure Key Vault

Sensitive information management patterns

- Create secure channels

- Encrypt data client-side

- Use bring your own key (BYOK)

Availability pattern

- Design for denial of service

Summary

Chapter 4 Threat modeling

- TL;DR

- What is threat modeling?

- The four main phases of threat modeling

- STRIDE's threat-classification approach

- The trouble with threat modeling

- Searching for a better threat modeling process

- A better way to perform threat modeling: The five factors

- Threat-modeling tools

- Assessing the five factors

- CAIRIS

- Microsoft Threat Modeling Tool

- OWASP Threat Dragon

- pytm

Table of Contents

Threatgile

Threats Manager Studio

How to threat model: A real-life example

Analyzing the solution: The first meeting

Analyzing the solution: The second meeting

Identifying specific threats and mitigations

Automatically identifying additional threats and mitigations

Creating the roadmap

Using the dashboard

Pushing selected mitigations into the backlog

Summary

Chapter 5 Identity, authentication, and authorization

Identity, authentication, and authorization through a security lens

Authentication vs. authorization vs. identity

Modern identity and access management

Identity: OpenID Connect and OAuth2 fundamentals

OpenID Connect and OAuth2

Application registration

Microsoft Authentication Library

OAuth2 roles

Flows

Client types

Tokens

Scopes, permissions, and consent

Anatomy of a JWT

Using OAuth2 in your Azure applications

Authentication

Something you know

Something you have

Something you are

Multifactor authentication

Who is authenticating whom?

Table of Contents

Creating your own authentication solution

The role of single sign-on

Getting access without authenticating

Authenticating applications

Authorization

Azure AD roles and scopes

Azure control plane built-in RBAC roles

Azure data plane built-in RBAC roles

Managing role assignments

Custom role definitions

Denying assignments

Role assignment best practices

Azure AD Privileged Identity Management

Azure attribute-based access control

Summary

Chapter 6 Monitoring and auditing

Monitoring, auditing, logging, oh my!

Leveraging the Azure platform

Diagnostic settings

Log categories and category groups

Log Analytics

Kusto queries

Raising alerts

Protecting audit logs

Using policy to add logs

Taming costs

The need for intentional security monitoring and auditing

The role of threat modeling

Custom events

Alerts from custom events on Azure Sentinel

Summary

Chapter 7 Governance

Table of Contents

Governance and the developer

Azure Security Benchmark version 3

- Network security
- Identity management
- Privileged access
- Data protection
- Asset management
- Logging and threat detection
- Incident response
- Posture and vulnerability management
- Endpoint security
- Backup and recovery
- DevOps security
- Governance and strategy

Governance enforcement

- Enforcement through processes
- Governance documentation and security education
- Role-based access control
- Automated enforcement during deployment

Microsoft Defender for Cloud

- Secure Score
- Reviewing compliance state for solution

Azure Policy

- Azure Initiatives and compliance frameworks
- Azure Policy effects
- Enforcement (effects) levels and RBAC by environment
- Policy Assignments
- Policy as code

Summary

Chapter 8 Compliance and risk programs

- Something important to get out of way
- What is compliance?

Table of Contents

HIPAA
HITRUST
GDPR
PCI DSS
FedRAMP
NIST SP 800-53
NIST Cybersecurity Framework
FIPS 140
SOC
ISO/IEC 27001
ISO/IEC 27034
Center for Internet Security Benchmarks
Azure Security Benchmark
OWASP
MITRE
Compliance synopsis

Using threat models to drive compliance artifacts

Summary

PART II: SECURE IMPLEMENTATION

Chapter 9 Secure coding

Insecure code

Rule #1: All input is evil

Verify explicitly

Determine correctness

Reject known bad data

Encode data

Common vulnerabilities

A01: Broken access control

A02: Cryptographic failures

A03: Injection

A04: Insecure design

A05: Security misconfiguration

Table of Contents

A06: Vulnerable and outdated components

A07: Identification and authentication failures

A08: Software and data integrity failures

A09: Security logging and monitoring failures

A10: Server-side request forgery (SSRF)

Comments about using C++

Don't write glorified C

Use compiler and linker defenses

Use analysis tools

Security code review

Keeping developers honest with fuzz testing

Generating totally random data

Mutating existing data . .

Intelligently manipulating data knowing its format

Fuzzing APIs

Summary

Chapter 10 Cryptography in Azure

A truth about security

Securing keys

Access control and Azure Key Vault

Use Key Vault Premium in production

Enable logging and auditing .

Network isolation

Use Microsoft Defender for Key Vault

Back up your Key Vault assets

Managed HSM and Azure Key Vault

Secure keys with Key Vault summary

Cryptographic agility

How to achieve crypto agility

Implement crypto agility

The Microsoft Data Encryption SDK

Optional parameters

Table of Contents

Managing SDK keys in Key Vault

Azure services and cryptography

Server-side encryption with platform-managed keys

Client-side encryption

Azure Storage cryptography

Azure VM cryptography

Azure SQL Database and Cosmos DB cryptography

Key rotation

Azure Key Vault key rotation

Protecting data in transit

TLS and crypto agility

Ciphersuites

TLS in Azure PaaS

Setting ciphersuites

TLS in Azure IaaS

A common TLS mistake in .NET code

Testing TLS

Debugging TLS errors

Unsecure use of SSH

Summary

Chapter 11 Confidential computing

What is confidential computing?

Confidential computing processors

Intel Software Guard Extensions

AMD Secure Encrypted Virtualization-Secure Nested Paging

Arm TrustZone

DCsv3-series VMs, SGX, Intel Total Memory Encryption, and Intel Total Memory Encryption

Multi-Key

Attestation

Trusted launch for Azure VMs

Azure Services that use confidential computing

Summary

Table of Contents

Chapter 12 Container security

What are containers?

You do not need containers for that!

How to proceed from here

Container-related services on Azure

Using containers on IaaS offerings

Comparing Azure container services

Problems with containers

Complexity

Immaturity

Fragmentation

Securing container services

Development and deployment

The container registry

The cluster

The nodes

The pods and containers

The application

Summary

Chapter 13 Database security

Why database security?

Which databases?

Thinking about database security

The SQL Server Family

SQL Server

Azure SQL Database

Azure SQL Managed Instance

Security in the SQL Server family

Control plane authentication

Control plane authorization

Control plane auditing

Control plane crypto on the wire

Table of Contents

- Control plane network isolation
- Data plane authentication
- Data plane authorization
- Data plane auditing
- Data plane crypto on the wire
- Data plane network isolation
- Cryptographic controls for data at rest
- Miscellaneous

Cosmos DB security

- Control plane authentication
- Control plane authorization
- Control plane auditing
- Control plane network isolation
- Data plane authentication
- Data plane authorization
- Data plane auditing .
- Data plane crypto on the wire
- Data plane network isolation
- Cryptographic controls for data at rest
- Miscellaneous

Encryption of data in use: Always Encrypted

- Always Encrypted
- Always Encrypted with secure enclaves
- Cosmos DB and Always Encrypted

SQL injection

Summary

Chapter 14 CI/CD security

What is CI/CD?

CI/CD tools

Source control systems and supply chain attacks

- Security tooling
- Protecting your developers

Table of Contents

- Pull request approvals and PR hygiene
- Separation of duties, least privilege overview

Secrets and service connections

Protecting the main branch in Azure DevOps and GitHub

Protecting the PROD deployment in Azure DevOps and GitHub

Securing deployment agents

- Securing Azure DevOps agents

- Securing GitHub agents

Summary

Chapter 15 Network security

Azure networking primer

- IPv4, IPv6 in Azure

- IPv4 concepts

- IPv4 addresses in Azure and CIDR

- Routing and user-defined routes

- Network security groups

- Application security groups

Landing zones, hubs, and spokes

- Hub and spoke and segmentation

- Environment segregation, VNets, and allowed communications .

- Ingress and egress controls

NVAs and gateways

- Azure Firewall

- Azure Firewall Premium SKU

- Azure web application firewalls

- API Management Gateway

- Azure Application Proxy

PaaS and private networking

- Private shared PaaS

- Dedicated PaaS instances

- Managed VNets

- Agent-based network participation

Table of Contents

Azure Kubernetes Service networking

- Ingress controls

- Egress controls with UDR

- Private endpoints for Kubernetes API server

- Cluster network policies

The dangling DNS problem

- An example

- Fixing dangling DNS

Summary

Appendix A: Core cryptographic techniques

Index