Repeating failure patterns – Challenges in enterprise cybersecurity

Prepared by: Boris So

Who am I?

Ethical Hacker?!

Security Architect?!

Security Researcher?!

Love coding and hacking!!

Love aviation!!

Love soccer, baseball, badminton!!

My Background

- Security/hacking researcher in non-profit organizations
  - Core member of VXRL
  - OWASP HK Chapter Lead

- Research interests
  - Obfuscated kernel rootkit: application in defeating anti-malware software
  - Steganography: application in anti-forensics
  - Machine learning and statistical modeling: application in detecting web attacks

What do I do?

- Day Job
  - Role
    - Cloud security consultant
    - Cybersecurity lead
  - Industry
    - Cloud service provider
    - Financial service institution
- Focus Areas
  - Secure programming
  - Ethical hacking
  - Exploit development
  - Protocol analysis
  - Computer forensics
  - Malware reverse engineering
  - Backdoor analysis

Fun Hacking Projects

AirGap

Adversarial Machine Learning

Cloud

Mobile

ICS

Web

IoT

Covert Channel

SDR/RF

Who are hackers?

Hackers

VS

Script Kiddies

How do hackers get in?

- Application vulnerabilities
- Platform vulnerabilities
- Malware

Past vs Present

- Observations in a few years back (Conceptual)
  - Old problem – Bad programming practice in web application development
  - New twist – Bad programming practice in mobile app development
  - Future forecast – Bad programming practice in IoT

Past vs Present

- Technical examples of repeating failure patterns (Low abstraction)
  - Old problem – HTTP response splitting in Java
  - New twist – HTTP response splitting in NodeJS

Past vs Present

- Technical examples of repeating failure patterns (Low abstraction)
  - Old problem – XSS in web sites
  - New twist – XSS in crypto-wallets

Past vs Present

- Technical examples of repeating failure patterns (Low abstraction)
  - Old problem – CSRF/XSRF in web applications
  - New twist – CSRF/XSRF in IoT devices

Past vs Present

- Technical examples of repeating failure patterns (Higher abstraction)
  - Old problem – Missing encryption in HTTP traffic (Web)
  - New twist – Missing encryption in Bluetooth Attribute Protocol / BLE communication (IoT)

Past vs Present

- Technical examples of repeating failure patterns (Higher abstraction)
  - Old problem – Open redirect (trusting untrusted URL) to redirect browser to malicious address
  - New twist – SSRF (trusting untrusted URL) to return data from internal service endpoint on cloud

Past vs Present

- Technical examples of repeating failure patterns (Highest abstraction)
  - Old problem – ROP (code reuse attack) in buffer overflow
  - New twist – POP (code reuse attack) in insecure deserialization

Why security failed?

- What did we do in the past?
  - Wrong priority
- Who did cybersecurity in enterprise?
  - Knowledge gap
  - Skill mismatch
- How did we implement defense?
  - Buy some toys
    - Firewalls, Scanners, Anti-Malware
  - Buy more toys
    - Find something else to buy

Technical

Why security failed?

- SME
  - Budget
  - Awareness
- Mega-Enterprise
  - Scale
  - Complexity

Money DOES Matter

Scale DOES Matter

Attack = Motive x Method x Weakness

Non-Technical

Why security still fails?

- Have we addressed the root cause of the issues?
  - Do we even know the problem?
- Have we changed the way we work?
  - Are we making conscious decisions?
    - Who should make the call?
- Are we repeating what we did in the past?
  - Is it because of uncontrollable constraints?
    - Are the constraints internal or external?

What had enterprises done?

- Has anything changed?
  - Assurance
    - We do security audit
    - We do penetration test
    - We do red team
  - Competence
    - We employ administrators
    - We employ risk managers
    - We employ penetration testers

Prospect in enterprise cybersecurity

- Enterprises are looking for people with
  - Deep technical skills
    - Who understands the limitations of technical defenses
    - Who can trouble-shoot technical problems
  - Good foundation of computer science and software engineering knowledge
    - Who understands how things work under the hood
    - Who can code, debug and build things
- At the same time
  - Good business sense
  - Strong communication and influence skills

Why don’t we just employ the top hackers?

Takeaways

- Understand the limitations of technical and non-technical security controls
- Human is still the weakest link but not just end-users
- No more buzz words in security products
- Don’t use English to defeat hackers

Always Deep Dive !!

Q&A