GSDC
Global Skill Development Council

ETHICAL HACKING REAL WORLD CASE STUDIES

www.gsdcouncil.org

HOW TO READ THESE CASE STUDIES

Each case study follows this structure:

What happened

The incident in plain language

How the attack worked

The phases and techniques used

What was missed

The security failures that made it possible

What an ethical hacker would have found

How a pen test could have prevented it

Key takeaways

What every ethical hacking professional must remember

CASE STUDY 1: The Yahoo Data Breach — The Largest Data Breach in History

What Happened

Between 2013 and 2014, attackers compromised Yahoo's systems and stole data belonging to all **3 billion user accounts** — every single Yahoo account that existed at the time. The breach was not discovered until 2016, and the full scope was not disclosed until 2017. At the time of disclosure, Yahoo was in the middle of acquisition talks with Verizon. The revelation cost Yahoo approximately **$350 million** in the final sale price.

How the Attack Worked

1

Reconnaissance

Attackers studied Yahoo's infrastructure, identified key employees with elevated system access, and mapped Yahoo's internal network architecture from publicly available information and earlier smaller probes.

2

Initial Access — Spear Phishing

Attackers targeted specific Yahoo employees with carefully crafted spear phishing emails — personalized messages referencing real projects, real colleagues, and real business context. At least one employee clicked, giving attackers their initial foothold.

3

Lateral Movement

From the initial compromised endpoint, attackers moved laterally across Yahoo's internal network — identifying systems with access to user databases, escalating privileges progressively.

4

Cookie Forging

The attackers discovered and exploited Yahoo's proprietary authentication cookie system. They were able to forge authentication cookies — accessing any Yahoo account without knowing the password. This enabled access at scale — billions of accounts without needing billions of passwords.

Data Exfiltration

User account data was exfiltrated over an extended period. Names, email addresses, telephone numbers, dates of birth, hashed passwords, and security questions were stolen.

Persistence — Staying Hidden for Two Years

The attackers maintained access to Yahoo's systems for approximately two years without detection. Yahoo's security monitoring failed to identify either the initial intrusion or the sustained presence.

Yahoo Breach — What Was Missed & What Ethical Hackers Would Have Found

What Was Missed

- Spear phishing susceptibility was never formally tested — employees had not been evaluated for social engineering vulnerability
- The cookie forging vulnerability in Yahoo's authentication system had never been identified through security testing
- Lateral movement across the internal network went undetected — no internal network segmentation prevented access from a single compromised endpoint to sensitive database systems
- Monitoring and detection systems failed to identify exfiltration of billions of records over an extended period
- Incident response was slow — the breach was active for two years before discovery

What an Ethical Hacker Would Have Found

A properly scoped penetration test would have tested employee susceptibility to spear phishing — revealing the human vulnerability that provided the initial entry.

An application security assessment would have identified the cookie forging vulnerability in Yahoo's authentication mechanism.

A network penetration test would have demonstrated the lateral movement paths that allowed an attacker to reach sensitive database systems from a compromised endpoint.

A detection and response test would have revealed that the monitoring systems were insufficient to identify sustained unauthorized access.

Yahoo Breach — Key Takeaways

1 **Spear phishing uses personalized information from reconnaissance**

Generic security awareness training does not prepare employees for targeted attacks.

2 **Authentication vulnerabilities — like cookie forging — are found through application security testing**

Not network scanning.

3 **A single compromised endpoint in a flat network gives an attacker access to everything**

Network segmentation limits blast radius.

4 **Breach detection capability is as important as breach prevention**

Two years of undetected access is a monitoring failure.

5 **The cost of a breach extends far beyond technical remediation**

$350 million in acquisition price reduction, plus regulatory fines, legal costs, and permanent reputational damage.

CASE STUDY 2: Target's Point of Sale Attack — How 40 Million Credit Cards Were Stolen Through an HVAC Contractor

What Happened

In the 2013 holiday shopping season, attackers stole credit and debit card information from approximately **40 million Target customers**. The attackers did not breach Target directly — they entered through a third-party HVAC (heating and air conditioning) contractor. The breach cost Target over **$200 million** in settlements, fines, and remediation costs, and contributed to the resignation of the company's CEO and CIO.

How the Attack Worked

01

Reconnaissance

Attackers identified that large retail organizations like Target often grant third-party vendors access to their networks for remote monitoring and management of building systems. They researched Target's vendor relationships and identified Fazio Mechanical — Target's HVAC contractor — as a potential entry point.

02

Initial Access — Vendor Compromise

Fazio Mechanical had network access to Target's systems for remote monitoring of refrigeration and HVAC equipment. Attackers compromised Fazio Mechanical's systems through a phishing attack targeting their employees. The credentials Fazio used to access Target's vendor portal were stolen.

03

Entry into Target's Network

Using Fazio's stolen credentials, the attackers accessed Target's vendor portal. Critically, this portal was not isolated from Target's payment systems — the vendor network was connected to the same network as Target's point of sale infrastructure.

04

Malware Deployment — RAM Scraping

Once inside the network, attackers deployed custom malware onto Target's point of sale terminals. The malware was designed to capture credit card data at the moment it was swiped — reading the card data from the terminal's memory before it was encrypted for transmission. This type of attack is called RAM scraping.

05

Data Collection & Exfiltration

The malware collected credit card data from swipes across Target's store network. The collected data was staged on a compromised internal server before being exfiltrated to external servers. Over 40 million card records were ultimately transferred outside Target's network.

06

Missed Detection

Target had deployed a security monitoring tool that actually detected the malware and generated alerts. Those alerts were reviewed by the security operations team in Bangalore — who escalated them. The escalated alerts were not acted upon by the US security team before the exfiltration was complete.

Target Breach — What Was Missed & What Ethical Hackers Would Have Found

What Was Missed

- Third-party vendor security was never formally assessed — vendors had access to Target's network without demonstrating adequate security controls
- Network segmentation was absent — a vendor portal for building management systems had a pathway to payment processing systems
- The principle of least privilege was not applied — Fazio Mechanical's access was not limited to only the systems required for HVAC monitoring
- Security alerts were generated but not acted upon — a process failure as much as a technical failure
- Point of sale terminal integrity was not monitored — malware on terminals went undetected for weeks

What an Ethical Hacker Would Have Found

A **third-party risk assessment** would have identified Fazio Mechanical as a high-risk vendor with network access but inadequate security controls.

A **network penetration test** would have demonstrated the lateral movement path from the vendor network to payment systems — a critical finding that should never exist.

A **social engineering test** of Fazio Mechanical's employees would have demonstrated phishing susceptibility.

A **point of sale security assessment** would have tested whether malware could be deployed on terminals and whether that deployment would be detected.

Key Takeaways

Third-Party Attack Surface

Third-party vendors are an extension of your attack surface — their security is your security problem.

Network Segmentation

Network segmentation between vendor access and sensitive systems is non-negotiable — what a contractor needs to see heating data has nothing to do with payment systems.

Least Privilege

The principle of least privilege means access is limited to exactly what is required for the specific function — not broad network access because it is convenient.

Alert Response

Security alerts that are generated but not acted upon represent a process failure — technology cannot protect an organization whose processes do not respond to it.

Supply Chain Attacks

Supply chain attacks are now one of the most common enterprise breach vectors — every major penetration test should include third-party access assessment.

CASE STUDY 3: Equifax Data Breach — An Unpatched Vulnerability That Exposed 147 Million People

What Happened

In 2017, Equifax — one of the three largest credit reporting agencies in the United States — suffered a breach that exposed the personal information of **147 million people**. This included names, social security numbers, birth dates, addresses, and in some cases driver's license numbers and credit card numbers. The data stolen was among the most sensitive personal information possible. The Federal Trade Commission fined Equifax **$575 million**. The breach was caused by a vulnerability that had a patch available for **two months** before the breach occurred.

How the Attack Worked

1 March 2017 — Vulnerability Disclosed

Apache Struts critical vulnerability CVE-2017-5638 disclosed — allowing remote code execution through a specially crafted HTTP header. A patch was released the same day.

2 Equifax's Failure to Patch

Equifax's security team was notified and instructed teams to patch within 48 hours. The consumer dispute portal — the system ultimately breached — was not patched. It ran the vulnerable Apache Struts version for **78 days** after the patch was available.

3 Exploitation

Attackers discovered the unpatched Equifax system and exploited CVE-2017-5638. The vulnerability allowed them to execute commands on Equifax's server without authentication — complete remote control of the system.

4 Lateral Movement & Data Discovery

From the initial compromised server, attackers explored Equifax's internal network over 78 days. They discovered databases containing consumer credit information and identified and queried **48 separate database tables** containing personal information.

5 Exfiltration

Data was exfiltrated in small encrypted batches to evade detection. Equifax's network monitoring could not inspect encrypted outbound traffic — the exfiltration was invisible. An expired SSL certificate had effectively blinded the security monitoring system for months.

6 Discovery

The breach was discovered when Equifax's security team renewed an expired SSL certificate that had been preventing inspection of encrypted traffic. When traffic inspection resumed, the malicious outbound connections were immediately visible.

Equifax Breach — What Was Missed & What Ethical Hackers Would Have Found

What Was Missed

- Patch management was inadequate — a critical vulnerability with a known patch went unpatched for 78 days on an internet-facing system
- Vulnerability scanning did not identify the unpatched system — or the results were not acted upon
- Network monitoring could not inspect encrypted outbound traffic — an SSL certificate had expired and was never renewed, leaving a blind spot in monitoring
- Database access controls were insufficient — attackers were able to query 48 different databases from a single compromised web server
- Outbound data transfer was not monitored for volume anomalies — significant data exfiltration over 78 days went undetected

What an Ethical Hacker Would Have Found

A **vulnerability assessment** would have identified the unpatched Apache Struts installation — CVE-2017-5638 was a publicly known critical vulnerability with scanning signatures available.

A **penetration test** would have confirmed exploitability and demonstrated the extent of access achievable from the vulnerable system.

A **network security review** would have identified the expired SSL certificate and the resulting monitoring blind spot.

A **database access control review** would have demonstrated that excessive database permissions allowed a web server compromise to escalate to database access.

Key Takeaways

→ **Patch management is a foundational security control** — a known, patchable vulnerability that is not patched is an invitation.

→ **Vulnerability scanning must be followed by remediation** — finding vulnerabilities and not fixing them provides false assurance.

→ **Monitoring blind spots are as dangerous as vulnerabilities** — an expired certificate that disables traffic inspection is a critical security failure.

→ **Database access from web servers should be limited to exactly the tables the application needs** — the principle of least privilege applied at the database level.

→ **The combination of a known vulnerability, poor patch management, inadequate monitoring, and excessive permissions** is how a preventable breach becomes a historic one.

CASE STUDY 4: Twitter's Social Engineering Attack — When VIP Accounts Become Weapons

What Happened

In July 2020, attackers compromised Twitter's internal administrative tools and took over the accounts of prominent verified users — including Barack Obama, Elon Musk, Jeff Bezos, Apple, Uber, and Joe Biden. The compromised accounts were used to post Bitcoin scam messages. Within minutes, the scam received over **400 Bitcoin transactions** — approximately **$120,000** in value at the time. The attackers were a group of young individuals, the oldest being 22 years old. **The attack was not sophisticated technically — it was entirely social engineering.**

How the Attack Worked

01

Reconnaissance

The attackers did not target Twitter's technical systems first. They targeted Twitter's employees. They researched which employees had access to internal administrative tools — specifically Twitter's "God mode" admin panel that allowed changing account email addresses, disabling two-factor authentication, and controlling account access.

02

Social Engineering — Vishing

The attackers called Twitter employees by phone, impersonating Twitter's IT department. They told employees they were calling about a VPN issue. They directed employees to a fake internal website designed to look like Twitter's actual VPN portal. Employees entered their credentials on the fake site — giving the attackers valid Twitter employee login credentials.

03

Internal Access

Using stolen employee credentials, the attackers accessed Twitter's internal administrative tools. They were able to look up any Twitter account, view associated phone numbers and email addresses, and make administrative changes.

04

Account Takeover

The attackers changed the email addresses and disabled two-factor authentication on targeted high-profile accounts — locking the real owners out and taking full control. They then posted the Bitcoin scam messages from those accounts to millions of followers.

05

Discovery and Response

Twitter identified the attack and responded by temporarily disabling the ability of all verified accounts to post — an unprecedented action that affected hundreds of thousands of accounts globally. The investigation revealed the social engineering vector within hours.

06

Arrest

The FBI identified and arrested three individuals, the youngest of whom was 17 years old. The primary attacker was 17. They were identified through cryptocurrency transaction tracing.

Twitter Attack — What Was Missed & What Ethical Hackers Would Have Found

What Was Missed

- Employee susceptibility to vishing attacks had never been formally tested
- There was no verification procedure for IT support calls — employees had no way to confirm the caller was genuinely from Twitter's IT department
- Administrative tools with the power to control any account on the platform were not protected by hardware security keys or out-of-band verification for sensitive actions
- A fake Twitter internal website was operational long enough to collect multiple sets of employee credentials without being detected
- The principle of least privilege was not applied to internal admin tools — more employees had access to more administrative functions than their roles required

What an Ethical Hacker Would Have Found

A **social engineering penetration test — specifically vishing** — would have revealed employee susceptibility to phone-based impersonation attacks.

A **physical security assessment** of internal systems access would have identified that sensitive administrative actions required insufficient verification.

An **employee security awareness assessment** would have identified that verification procedures for IT support contacts were absent or inadequate.

An **administrative access review** would have identified excessive access to powerful internal tools that did not require that level of access for their roles.

Key Takeaways

Phone Call Bypass

The most sophisticated technical security can be bypassed by a convincing phone call.

Vishing Underestimated

Vishing — voice phishing — is consistently underestimated as an attack vector compared to email phishing.

Verification Procedures

Employees must have a clear, practiced procedure for verifying the identity of anyone requesting sensitive actions or credentials — especially unsolicited contacts.

Stronger Admin Authentication

Internal administrative tools with platform-wide power require stronger authentication than standard employee accounts — hardware security keys and out-of-band verification for sensitive actions.

People Skills, Not Coding Skills

Age and technical sophistication are not correlated in social engineering attacks — the technique requires people skills, not coding skills.

CASE STUDY 5: The Colonial Pipeline Ransomware Attack — One Leaked Password Shuts Down a Nation's Fuel Supply

What Happened

In May 2021, Colonial Pipeline — which operates the largest fuel pipeline in the United States, supplying approximately **45% of fuel to the US East Coast** — was forced to shut down operations following a ransomware attack by the criminal group **DarkSide**. The shutdown lasted **six days**, causing fuel shortages, panic buying, and price spikes across the southeastern United States. Colonial Pipeline paid **$4.4 million in ransom** to receive a decryption key. The US Department of Justice later recovered approximately **$2.3 million** of that payment by tracking the cryptocurrency wallet.

How the Attack Worked

1

Initial Access — Compromised Credentials

Investigators found that the attackers gained access through a single compromised set of VPN credentials. The password for a VPN account had been exposed in a previous, unrelated data breach — likely discovered through dark web monitoring or credential trading. The VPN account did not have multi-factor authentication enabled.

2

No Multi-Factor Authentication

The VPN account used only a username and password. When the attackers obtained the password — from the earlier data breach — they could log directly into Colonial Pipeline's network with no additional verification required.

3

Ransomware Deployment

Once inside the network, the attackers moved laterally, identifying and accessing Colonial Pipeline's business systems. They deployed DarkSide ransomware across the business network — encrypting files and data critical to operations.

4

The Shutdown Decision

Colonial Pipeline's operational technology (OT) systems — the systems that actually control the physical pipeline — were not directly compromised by the ransomware. However, Colonial Pipeline made the decision to shut down pipeline operations proactively because their billing and business systems were encrypted — they could not determine how much fuel had been delivered and therefore could not bill customers. **A billing system failure shut down critical infrastructure.**

Ransom Payment

Faced with an extended shutdown and unable to restore systems quickly, Colonial Pipeline paid the $4.4 million ransom in Bitcoin to receive the DarkSide decryption key. The decryption process was described as slow — restoration took days even after the key was received.

Colonial Pipeline — What Was Missed & What Ethical Hackers Would Have Found

What Was Missed

- A single VPN account without multi-factor authentication was accessible from the internet
- Credential monitoring was not in place — the compromised password had been exposed in a previous breach but was still in use
- Password reuse policies were not enforced — employees were not required to use unique passwords or check credentials against known breach databases
- Network segmentation between business systems and operational technology was insufficient — the business system compromise directly impacted operational decisions
- Backup and recovery procedures for encrypted systems had not been tested — restoration was slower than the attacker's own decryption tool

What an Ethical Hacker Would Have Found

An **external attack surface assessment** would have identified internet-facing VPN accounts without multi-factor authentication — a critical finding.

A **credential audit against known breach databases** would have identified that active credentials had been compromised in previous breaches.

A **network segmentation review** would have identified the connectivity between business systems and operational decision-making that allowed a business system attack to shut down physical infrastructure.

A **business continuity test** would have revealed the dependency on billing systems for operational decisions and the slowness of recovery procedures.

Colonial Pipeline — Key Takeaways

MFA Is the Highest-Impact, Lowest-Cost Control

Multi-factor authentication on all remote access systems is the single highest-impact, lowest-cost security control — one missing MFA instance shut down critical national infrastructure.

Credential Exposure Enables Cross-System Attacks

Credential exposure in one breach enables attacks on other systems — active credentials must be checked against known breach databases regularly.

Understand Business-to-OT Dependencies

Critical infrastructure operators must understand the dependencies between business systems and operational decisions — a billing system failure should not shut down a pipeline.

Ransomware Recovery Requires Tested, Offline Backups

Paying the ransom is expensive, uncertain, and funds criminal organizations.

Operational Resilience Is as Important as Technical Security

The ransom payment decision was driven by business continuity failure, not inability to remove the malware.

Cross-Case Patterns — What Every Ethical Hacker Must Know

Across all five case studies, the same categories of failure appear repeatedly. These are not isolated mistakes — they are systemic vulnerabilities that ethical hackers are trained to find.

Human Vulnerability

Yahoo, Target, Twitter, and Colonial Pipeline all had a human element at the point of initial access. Spear phishing, vishing, and vendor compromise all exploit people — not just technology. Social engineering testing is not optional.

Flat Networks Enable Lateral Movement

In every case, attackers moved from an initial foothold to sensitive systems because network segmentation was absent or inadequate. A compromised endpoint should never have a direct path to a payment system, a user database, or operational technology.

Unpatched Systems Are Open Doors

The Equifax breach was entirely preventable — a patch existed for 78 days before exploitation. Vulnerability management is not a one-time scan; it is a continuous process with mandatory remediation timelines.

Detection Failures Extend Dwell Time

Yahoo's attackers stayed for two years. Equifax's exfiltration ran for 78 days. Target's alerts were ignored. Detection capability — and the processes to act on it — is as important as prevention.

The Ethical Hacker's Toolkit — Methods That Would Have Prevented These Breaches

Assessment Type	What It Tests	Breach It Would Have Prevented
Spear Phishing Simulation	Employee susceptibility to targeted, personalized phishing emails	Yahoo (initial access), Target (vendor compromise)
Vishing Test	Employee susceptibility to phone-based impersonation attacks	Twitter (credential theft via fake VPN portal)
Network Penetration Test	Lateral movement paths from compromised endpoints to sensitive systems	Yahoo, Target, Equifax, Colonial Pipeline
Application Security Assessment	Authentication vulnerabilities, cookie forging, injection flaws	Yahoo (cookie forging), Equifax (Apache Struts RCE)
Vulnerability Assessment	Known unpatched vulnerabilities on internet-facing systems	Equifax (CVE-2017-5638 unpatched for 78 days)
Third-Party Risk Assessment	Vendor access controls, vendor security posture	Target (Fazio Mechanical), Colonial Pipeline (VPN credentials)
Credential Audit	Active credentials exposed in known breach databases	Colonial Pipeline (reused breached password)
Administrative Access Review	Excessive privileges on internal tools and systems	Twitter (God mode admin panel), Equifax (database permissions)
Detection & Response Test	Monitoring capability, alert response processes	Yahoo (2-year dwell time), Target (ignored alerts), Equifax (expired SSL)
Business Continuity Test	Recovery procedures, backup integrity, OT-IT dependencies	Colonial Pipeline (billing dependency, slow decryption)

Final Principles — What Every Security Professional Must Carry Forward

These five case studies — Yahoo, Target, Equifax, Twitter, and Colonial Pipeline — represent some of the most consequential security failures in modern history. Together, they teach the same lessons.

People Are the Perimeter

Technical controls are bypassed through human vulnerability. Social engineering testing — phishing, vishing, and impersonation — must be part of every security program.

Segment Everything

A single compromised endpoint should never reach sensitive data. Network segmentation and the principle of least privilege limit blast radius when — not if — a breach occurs.

Patch Without Exception

Known vulnerabilities with available patches are not acceptable risks. Patch management must be enforced with mandatory timelines and verified completion.

Detect and Respond

Prevention is not enough. Monitoring must be continuous, complete, and connected to processes that act on alerts. Dwell time is the measure of detection failure.

MFA Is Non-Negotiable

Multi-factor authentication on all remote access is the single highest-impact, lowest-cost control available. One missing MFA instance shut down critical national infrastructure.

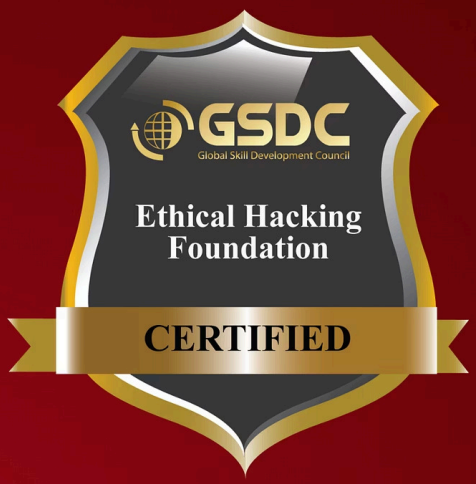
Your Vendors Are Your Risk

Third-party access is an extension of your attack surface. Every vendor with network access must be assessed, monitored, and held to the same security standards as internal systems.

The ethical hacker's role is not to break things — it is to find what attackers will find before attackers find it. Every vulnerability identified in a penetration test is a breach that did not happen. Every control validated is a risk that was managed. These case studies exist so that the next breach does not.

GSDC
Global Skill Development Council

CERTIFIED ETHICAL HACKING FOUNDATION (CEHF)



ABOUT GSDC CERTIFICATION

EBOOK

Extensive and exclusive Ebook created by world's experts to help you with understanding core concepts.

LEARNING MATERIALS

Get access to learning materials such as videos, ebooks, templates, and practice exams, which will help you clear the certification exam.

CREATED BY EXPERTS

GSDC certifications are created and authored by world's leading experts in the field.

LEARNING OBJECTIVE

- Gain insights into autonomous decision-making processes
- Apply knowledge using ready-to-implement templates
- Demonstrate ability to work with Agentic AI models
- Validate your skills wit

Enroll now with the code **LEARN20** To avail **20%** discount

Enroll Now

www.gsdcouncil.org