Hackers: Analysis of Attack Anatomies

T. Harrington

Independent Security Evaluators, USA

ABSTRACT

Whether to obtain pre-theatrical content assets from US studios or circumvent distribution rights of European broadcasters, hackers are highly motivated to attack the global media and entertainment industry. These attacks are facilitated by the current rapid adoption of embedded systems, cloud solutions, and web based platforms. These attacks often undermine the very collaboration, cost-efficiency, monetization, scalability and user experience goals for which these systems were designed and deployed. As malicious hackers advance their techniques at a staggering pace, often rendering current defense tactics obsolete, so too must security practitioners obsess over deploying progressive techniques. Presented by the elite American organization of white hat hackers most widely known for being first to break the iPhone and the only security consulting firm engaged in the security team of USC's Project Cloud initiative, this paper analyzes the anatomies of real world attacks against high profile systems. It will extract lessons from these attack anatomies to provide a framework to account for these modern attackers, articulate context to the global media and entertainment industry, and supply readers with key takeaways, including immediately actionable guidance.

INTRODUCTION

In the current digital era, executives leading companies of all sizes are facing a daunting challenge in defending their most valuable digital assets. Modern adversaries are very sophisticated, attack vectors are ever evolving, and digital assets are becoming exponentially more valuable. Traditional defenses alone are no longer effective against these adversaries. However, Chief Executive Officers and the executives who support them should not lose hope, as there are techniques that all companies can adopt in order to more effectively protect their assets in such a complex defense landscape. These techniques are realistic to implement and in many cases are more cost-efficient than the lesser-effective traditional approaches. In this paper, we investigate a series of high profile breaches in order to understand the anatomy of each attack, and then extract security lessons from there.

Independent Security Evaluators

In 2005, three PhD candidates and one professor of the Information Security Institute of Johns Hopkins set up a lab to study RFID devices, understanding that there might be commercial interest if they were successful in breaking some high-profile systems. The team started with the Texas Instruments Digital Signature Transponder (DST). This was chosen for two reasons: First, at the time this was considered "unbreakable." Second, this system powered two very important and high profile use cases: the immobilizer function of

Ford Motor Company ignition keys, which is an electronic prevention measure against forged keys starting automobile ignitions; and the Exxon Mobile SpeedPass™, a dongle attached to the user's keychain and linked to the user's credit card. An attack on either system carries obvious implications for theft, brand reputation, and personal safety**.**

It took the team two weeks to reverse-engineer the cryptic algorithm, a few more to create a non-functioning prototype, and another few weeks to create a fully functional radio. To prove concept, the team invited several news outlets to watch a demonstration in which the team started a Ford with a key the reporters had watched the team make at Lowes Hardware, started the car, drove to an Exxon Mobile station, and pumped free gas. Their success gained national press and commercial interest, thus beginning Independent Security Evaluators (ISE). Today ISE has grown into a very sophisticated commercial enterprise-class consulting firm, dividing our time between research like that of the Texas Instruments case study, wherein we try to find the vulnerabilities of a system in order to advance some particular cause, and working directly with companies who hire us to find all ways in which an interested adversary could compromise their systems and to help develop mitigation strategies.

CASE STUDY: TARGET

Over the all-important holiday shopping period of Q4 2013, cyber thieves broke into American retail giant Target and installed malware which resulted in the theft of 40 million credit card numbers and an additional 70 million accounts of customer information. To begin the attack, the hackers used a spear-fishing campaign to obtain some credentials of Target's HVAC vendor. The vendor had remote access to Target's network environment for reasons of monitoring energy consumption and temperature control, yet it also had access to the payment environment. Therefore, once in the system the attackers used the authentic credentials gained through the vendor to jump from the maintenance environment to the payment environment. There they installed ram-scraper malware which copied digits resembling credit card numbers to text file. Once they obtained the growing data file, the attackers employed an exfiltration system whereby they leveraged NetBIOS and used communication to move the files to an area in the network where they could be removed without alarm. Once removed, the files were dumped to compromised servers around the world for the attackers' retrieval.

Profit loss was immense. During the crucial two month holiday following the breach, profit fell 46%, or $441 million (The Associated Press, 2014) and within eight months of the breach response costs already sat at $236 million (Circa 1605, Inc. 2015). According to investment analysis (Sksriachan and Finkle, 2014) nearly every key investor metric at Target was down in Q4 2013, causing performance to fall short of projections: transaction count decreased 5.5% (a rate surpassing even the 4.8% decline at the peak of the 2008 financial crisis), sales decreased 3.8%, and sales at stores open at least a year fell 2.5%. Over one year later the fallout for Target is still unfolding: analysts expect the damages could soar to $5.2 billion or more.

Key Takeaway: Secure Assets, not Perimeters

Traditionally, defenses have focused on hardening perimeters. Though worthwhile, this no longer works with modern attacks as modern attacks happen from within trusted boundaries. In today's environments, third party systems are so integrated into an

infrastructure that there is no longer a clear distinction between “internal” and “external.” Such as the case with Target, a modern attacker will compromise a system through a stepping stone, whereby they attack a smaller vendor associated with the large company and use the vendor’s insecurities to infiltrate the company. Rather than securing the perimeter, layers of defense in depth can secure individual assets. Layered defense allows executives to consider what adversaries could access were they inside the environment today, and firewall off the most valuable assets from other assets, making it difficult for adversaries to move from one asset to the next. For Target, hardening applications, infrastructures, and the supply chain is an effort estimated to cost in the low single digit millions – but in exchange would have saved what has already cost over $61 million in response costs, plus $441 million in lost Q4 income, plus further yet-unknown billions in possible punitive damages.

CASE STUDY: CHIPSET MANUFACTURER (UNDISCLOSED)

A major secure chipset manufacturer recently contacted ISE to perform a black box penetration test of its flagship product. Their stated objective was to determine the likelihood of an external adversary thwarting their defenses and compromising their most valuable assets. Considering this goal, we strongly advised that a black box penetration test would not adequately meet the client’s objective, and that they instead perform a white box vulnerability assessment. After much deliberation, we arrived at a compromise whereby we performed *both* a black box and white box test, which ultimately enabled us to arrive at a side-by-side quantifiable comparison of the two approaches. To compare the two test ideologies, ISE allocated equal resources to each assessment round and performed them in series: two months of a black box test followed by two months of a white box test.

The differences between black box penetration testing and white box vulnerability assessment can be broken down into two sections: methodology (black box vs. white box), and evaluation (penetration testing vs. vulnerability assessment). The differences in the methodology of a black or white box assessment come down to knowledge. In a *white box* assessment, the evaluator has full detailed knowledge of system functionality. In a *black box* assessment, the evaluator has very limited knowledge, obtaining information only from outputs that result from varying test inputs, and with no knowledge about the inner workings of the system. Comparing evaluative processes, the objective of a *vulnerability assessment* is to determine the full scope of exposures that exist—quite simply, a vulnerability assessment is a risk assessment. Vulnerability assessments seek to identify all ways in which asset compromise might be possible. The goal of a *penetration test* is simply to determine if defenses can be breached. In terms of risk assessment, it provides primarily a binary risk rating: either the defenses can or cannot be breached. After comparing methodology and evaluation, it can be seen that the most effective calculation of risk is derived from the combination of white box methodology and vulnerability assessment.

Key Takeaway: White Box Vulnerability Assessment Over Black Box Penetration Test

Figure 1—Cost effectiveness of black box penetration testing versus white box vulnerability assessment.

Over the period of two months and 200 hours' work using the black box penetration test, we discovered four (4) potential security vulnerabilities. Of those four (4) vulnerabilities: two (2) the customer already knew about, and so time and money were wasted re-investigating known issues; a third issue was a misunderstanding by us about how the system worked (remember: with black box methodology, the assessor has no inside knowledge of how the system functions); and ultimately one (1) issue was confirmed as severe and previously unknown. No other issues were discovered by the black box test. Furthermore, no mitigation strategies were provided, due to the lack of system knowledge inherent with a black box methodology. By contrast, over the same period of time, the white box assessment uncovered eleven (11) confirmed severe issues, ten (10) confirmed other issues, and devised over 21 mitigation strategies for the company. ISE was substantially more confident of the white box findings than the black box.

By placing a dollar value to man hours spent, the cost effectiveness of a white box vulnerability assessment becomes clear. The difference in cost per issue is staggering: 200 hours per issue with a black box versus ~9 hours per issue with a white box.

CASE STUDY: BELKIN

Despite having known insecurities, small office/ home office (SOHO) networking equipment has received surprisingly little attention from security researchers. As a research study, ISE set out to identify vulnerabilities within common routers, with a goal to present manufacturers with the findings aiming to drive change within the industry and to protect and empower consumers. The objective was to study the 10 most popular SOHO routers and to identify issues within one-third. The scope of the project was expanded to the top 13 routers, with staggering results: ISE found that all 13 routers evaluated could be taken over from the local network, with four (4) of the attacks requiring no active management session. Eleven (11) of thirteen (13) routers evaluated could be taken over from the wide area network (WAN), and two (2) of these attacks required no active management session.

The data shows that each was discovered to be susceptible to a remote adversary (in which an attacker from somewhere outside the victim's vicinity breaks into the router by way of a malicious link or something similar), a local adversary (a potentially more harmful situation than the remote attack: a point-click-kill operation wherein an attacker within the victim's vicinity breaks into public wifi and runs an automatic attack), or both. However,

after going through responsible disclosure with all the manufactures, no changes have been implemented. Consequently, a worst-case scenario presented itself due to these issues: in March 2014, hackers took control of over 300,000 home routers.

Key takeaway: Security vs. Functionality

Router manufacturers have prioritized functionality over security. A consistent business-wide theme is that security and functionality should be separated across all systems, presenting a lesson that boils down to conflict. Conflict between teams creates a healthy environment of balance. As an example of healthy conflict, examine the relationship between a Chief Marketing Officer and a Chief Financial Officer. A CMO may decide to spend $100,000 on conference expenses, and without a CFO to question the expenses the CMO spends the money. The purpose of the CFO in this situation is to ask the CMO to prove that the expenses are the best use of money for the customers and for the business. The conflict between teams creates a dialogue which theoretically derives the best outcome for the organization.

With technology, security and functionality have different priorities. Security's priorities lie within protecting assets, assessing access control, and defense in depth. Functionality prioritizes user experience, timely delivery, and overall performance. None of the priorities overlap. When the two are placed within the same team, prioritization naturally happens, but nine times out of ten functionality's priorities trump security. By separating the two into different factions, the priorities of each faction need not be compromised. In addition to increasing bandwidth to each section, separating security and functionality creates a healthy conflict between the two, helping each to keep progress and priorities in check.

CASE STUDY: SNAPCHAT

SnapChat was hacked in December of 2013. The subject of the attack was its Find Friends feature: Find Friends uploads all of a user's contacts to Snapchat, and Snapchat then sends back all user information from within the contacts uploaded. Because of the rush to release the feature the company failed to put on the proper rate-limit protection. Therefore, the attackers were able to upload a massive database of all the phone numbers within an area code, gaining access to all user information pertaining to each number uploaded. The stolen data was then uploaded to a site called SnapChatDB.info and made available for download (Shu, 2013).

Key Takeaway: Build Security In, Do Not Bolt It On

Companies are often in such a rush to release the product or upgrade the new iteration that security becomes its last step. While the attackers claim the hack was to prove a security point, the information leaked onto SnapChatDB.info created a dangerous situation for the affected victims. Potential attackers could use the leaked usernames to gain actual names and phone numbers, giving them a powerful set of data to go social engineer any other account.

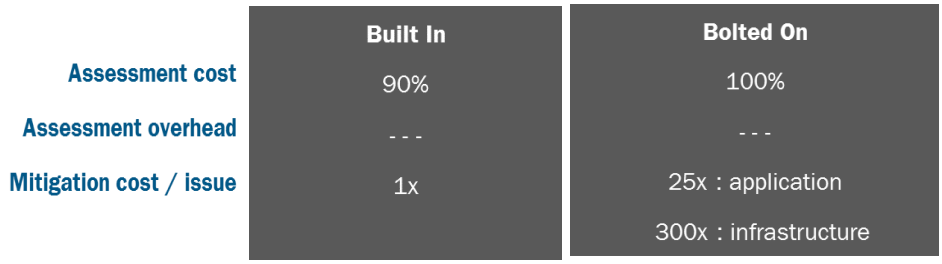

Figure 2—Comparison of assessment and mitigation costs taken from ISE case study

From a financial perspective, building security in during the development process is less expensive and far more effective than bolting it on at the end. To discuss the real financial ramifications of building in security, ISE aggregated and anonymized all metrics from our own customers over the past nine years of business and proved that it is less expensive to build security in. The savings happen in two areas. The first is assessment cost: if a company hires an organization such as ISE to help harden a system during each production process, it costs a company about 90% to build security in over the life of that process versus 100% of the cost of hiring someone to do a security review at the end. The second, and significantly larger area of savings is the remediation cost. When applications had bolted security on at the end, it cost the company about 25x more to fix its issues than had they been addressed during the development process. Likewise, we found that an infrastructure issue cost 300x more to fix at the end of the process.

CASE STUDY: iPHONE

The Apple iPhone was released to much fanfare on June 29, 2007. Because of its immense popularity and the large amount of personal information stored on these mobile devices, ISE decided to conduct a security analysis of the product. We hypothesized that the mobile experience, one of the crucial user experiences upon which the iPhone improved, was likely the existing Safari desktop browser migrated to a mobile version. In the weeks leading up to the release, we studied all of the vulnerabilities that were being disclosed about Safari and identified a few that seemed low enough within Apple's developers' priority queue that they might not be addressed in the mobile version. Therefore when the iPhone came out, one of the scripts we had prepared to run against the system worked: a buffer overflow attack took administrative control over a user's phone through a malicious website set up by ISE. Through the attack we could do everything the phone could do, including add and delete contacts, modify pictures, and send and receive text messages and emails. We proved concept through a reporter from the New York Times with whom we working: after willingly visiting the malicious website and subjecting his phone to the buffer overflow attack, we were able to send texts from our lab in Baltimore using his phone in New York.

Key Takeaway: Security Is an Ongoing Process

Overtime a system gains in complexity. Our attack strategy against Apple keyed in on the fact that there might be some issues in the development process that were not being addressed at appropriate intervals. ISE leveraged that by taking known information and creating an attack. A traditional way of looking at security assessment is that as a company iterates a system it will have security come back and be part of the process

periodically over time. A typical assessment cycle is about one year, but in some cases it is as infrequent as every two years. Yet, after the review has happened, issues have been mitigated, and development continues, the security baseline does not change but the attack surfaces do. A quarterly review cycle, or at most every six months, will allow security to expand alongside the ever-growing attack surfaces.

	Yearly	Bi-yearly	Quarterly
Initial assessment cost	X	X	X
Full scope reassessment cost	90-95%	35-45%	20-30%
Full assessments / year	1	2	4
Cost / year	X (0.9)	X (0.7)	X (0.8)

Figure 3—ISE's comparison of costs between yearly, bi-yearly, and quarterly security assessments

It would be easy to conclude that quadrupling the security assessment would quadruple the security costs. However, by looking through both our own data and publically available information ISE found that it is less expensive to get four reviews per year than to do one per year. One yearly full-scope reassessment costs 90-95% of the initial assessment, providing the company one (1) security assessment each year at nearly 100% the initial cost. However, each quarterly full-scope reassessment costs 20-30% of the initial assessment cost, therefore bringing the yearly assessment cost to roughly 80% of the initial cost and providing the company four (4) total reassessments. Money is saved on each assessment by streamlining the process through the elimination of the learning curve.

SUMMARY

To protect themselves against the scenarios discussed above, ISE always recommends that its customers arm themselves with key questions to ask of their vendors, and recommends that vendors be prepared to answer such questions as part of their sales pitch when trying to sell into a company. These questions could include:

How have you considered my assets in your system? Tell me about the threat model and the adversaries you are taking into account.

What design principles have you built into your development methodology? Explain your defense in depth, your use of the principle of least privilege, etc.

What sort of attack surfaces does your system introduce to my company? Remember, there is nothing wrong with admitting that a feature is an attack surface; in fact, ignoring that a feature is an attack surface is a weakness.

What sort of testing has been done during development and at what intervals? Has the company been doing only the necessary scans, and if so has it done anything with the reports? Has it done the more intensive manual white box-type reviews which identify and mitigate vulnerabilities most effectively?

What sort of review are you doing at an ongoing basis? How often, and what type?

Always remember: Secure assets, not simply perimeters. White box vulnerability assessments are a more thorough and cost effective testing method than black box penetration tests. Build security in during the development process, do not bolt it on at the end. Security should remain separate from functionality, and is an ongoing process. These key takeaways can help ensure the best security relationship between your company, your vendors, and your architects.

REFERENCES

1. Circa 1605, Inc. 2015. Target agrees to $19 million data breach settlement with Mastercard. Circa Business & Economy. http://cir.ca/news/target-stores-hacking-investigation

2. Shu, C. 2013. Confirmed: Snapchat Hack Not A Hoax: 4.6M Usernames And Numbers Published. Tech Crunch. http://techcrunch.com/2013/12/31/hackers-claim-to-publish-list-of-4-6m-snapchat-usernames-and-numbers/

3. Sksrichan and Finkle. 2014. Target shares recover after reassurance on data breach. Reuterse Technology. http://reuters.com/article/2014/02/26/us-target-results-idUSBREA1P0WC20140226

4. The Associated Press. 2014. Data breach costs take toll on Target profit. CBS Money Watch. http://www.cbsnews.com/news/data-breach-costs-take-toll-on-targe