# Software Security Paper

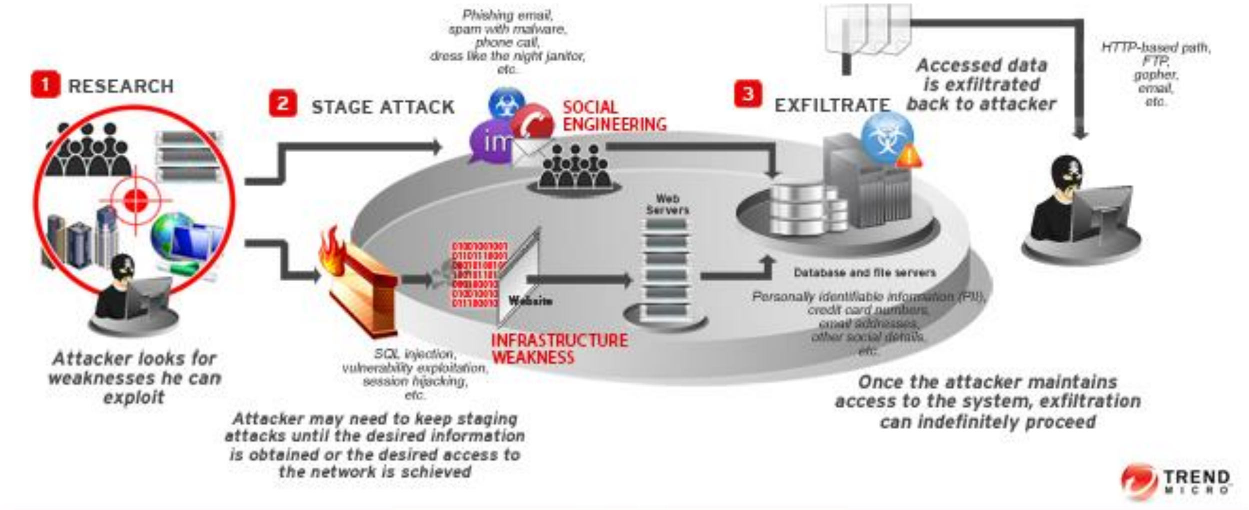
**By: Steven Davis**

**Class: CSC 424 Software Engineering 2**

**Due Date: April 23, 2021**

Since the advent of mass-produced software and global information exchange, there have been those that have sought to gain unauthorized access to pieces of software that could contain sensitive data. These kinds of attempts have led to the creation of modern-day malware, ransomware, and other kinds of computer viruses. In recent years, there have been many cases in which software was breached which led to personal information getting into unauthorized people's hands. To name a few, there were data breaches from Zoom and Nintendo back in April of 2020, data breach on Activision in September of 2020, and even an attack on Spotify in December of 2020[1]. In response to the ever growing threat of data breaches, programmers have had to build their software to be as secure as possible.

To start with, let's go over how a data breach or unauthorized access even occurs. For the most part, when someone gains unauthorized access to some software, they exploited a vulnerability in the software to some degree. This can be anywhere from just brute-force spamming data at a database until you manage to bypass the security, to intricately weaving your way through a few previously unknown flaws in the code to stealthily gain access to the data without anyone ever knowing. In either case, a person is using some kind of weakness that was discovered in the software in order to gain access to restricted information[2].


Phishing email,
spam with malware,
phone call,
dress like the night janitor,
etc.
1 RESEARCH
2 STAGE ATTACK
SOCIAL ENGINEERING
3 EXFILTRATE
Accessed data is exfiltrated back to attacker
HTTP-based path,
FTP,
gopher,
email,
etc.
Web Servers
Website
Database and file servers
Personally identifiable information (PII),
credit card numbers,
email addresses,
other social details,
etc.
INFRASTRUCTURE WEAKNESS
SQL injection,
vulnerability exploitation,
session hijacking,
etc.
Attacker looks for weaknesses he can exploit
Attacker may need to keep staging attacks until the desired information is obtained or the desired access to the network is schieved
Once the attacker maintains access to the system, exfiltration can indefinitely proceed
TREND MICRO

MALICIOUS DATA BREACH DIAGRAM

The image above shows the general flow of how a data breach occurs. First, the attacker will do his research to find a weakness in either the physical or digital side of the software. After that, they will stage their attack, using whatever vulnerability they find. Finally, they will access

[1] Bekker, E. (2021, February 22). 2020 Data Breaches - The Most Significant Breaches of the Year: IdentityForce®. We Aren't Just Protecting You From Identity Theft. We Protect Who You Are. https://www.identityforce.com/blog/2020-data-breaches.

[2] *Data Breaches 101: How They Happen, What Gets Stolen, and Where It All Goes*. (2018, August 10). Security News. https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/data-breach-101

the data they came after and quickly leave. These kind of attacks can last anywhere from a few seconds to a few hours depending on what the vulnerability found is and how stealthy the attacker can be.

While it is virtually impossible to create a software that has absolutely not vulnerabilities, there are ways that programmers can mitigate the risk of a vulnerability that can be exploited to the degree of a data breach. One of the easiest way is to, funny enough, hire a hacker to test your system. Surprisingly enough, there are people out there who you can pay to try and hack your system, and they will do so without taking anything private. These are commonly referred to as "bug bounty hackers[3]" and are used by many of the largest companies. Microsoft, Google, Tesla, and even the Department of Defense hire such hackers to test their systems[4]. What is great about people like this is that, while they will come at you with the same tenacity that your general hacker will, they will not take anything, and will immediately report the vulnerability. Sometimes, they may even have suggestions on how to deal with that vulnerability.

Once a vulnerability or multiple vulnerabilities have been found in the software, the next step is for the programmer(s) to develop a patch for the vulnerabilities and get it pushed out as soon as possible. Keeping on a fairly regular update schedule can help to thwart hacking attempts, as the vulnerability that a given hacker was working to exploit may be patched out fairly quickly[5]. After a patch is deployed, it may be wise to go back to the bug bounty hackers and let them have another go at the software. By using this simple loop, a company should be able to avoid any major data breaches.

Another fairly simple way to increase your software's security is to give very minimal access to the end user. By only giving the bare necessities needed for the end user, this gives a would-be attacker very few points of which to hit your software. This can range from assigning "levels" of your program to different people, to having the program transmit data over an encrypted server to the main program behind many layers of security in order to process.

While the previous items are more on the side of the programmer, there are things that can be done on the side of the company as a whole. Most importantly is creating a plan for what to do once an attack or data breach has been detected. This can allow for damage to be mitigated or outright nullified. For instance, if there is an attack detected but it has yet to reach the database, a smart thing to do, if possible, would be to cut the connection to the database as a whole. While this could cause some temporary issues for legitimate end users, it is better to have

[3] Wikimedia Foundation. (2021, March 24). *Bug bounty program*. Wikipedia. https://en.wikipedia.org/wiki/Bug_bounty_program.

[4] Fazzini, K. (2018, December 13). *Some freelance hackers can get paid $500,000 a year to test defenses of companies like Tesla*. CNBC. https://www.cnbc.com/2018/12/12/freelance-hackers-get-paid-to-test-the-defenses-of-firms-like-tesla.html.

[5] Synopsys Editorial Team. (2020, July 6). *Top 10 software security best practices: Synopsys*. Software Integrity Blog. https://www.synopsys.com/blogs/software-security/top-10-software-security-best-practices/.

a slight annoyance than for the database to be breached and private information leaked to the public. On the other side, if it has been found that a data breach has already occurred and completed, one of the first things that should be done is to find exactly what data has been affected. Once that has been done, it is important that the company quickly patch the vulnerability, then issue a statement about the breach. By issuing a statement, this allows for end users effected by the breech to take appropriate action, such as changing passwords or cancelling credit cards that have been leaked, while also possibly alerting others who may have the same or similar systems to work on deploying a patch to their systems.

In the end, software is a constant cat and mouse game between the programmer and hacker. No software will ever be 100% safe, but there are steps that can be taken in order to mitigate the risk. In the event that software was compromised, it is very important to patch the vulnerability as soon as possible and issues a statement of what occurred to alert both end users and other programmers as to what has occurred. All one can hope is that what they do is enough.

# APPENDIX

Bekker, E. (2021, February 22). 2020 Data Breaches - The Most Significant Breaches of the Year: IdentityForce®. We Aren't Just Protecting You From Identity Theft. We Protect Who You Are. https://www.identityforce.com/blog/2020-data-breaches.

*Data Breaches 101: How They Happen, What Gets Stolen, and Where It All Goes*. (2018, August 10). Security News. https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/data-breach-101

Fazzini, K. (2018, December 13). *Some freelance hackers can get paid $500,000 a year to test defenses of companies like Tesla*. CNBC. https://www.cnbc.com/2018/12/12/freelance-hackers-get-paid-to-test-the-defenses-of-firms-like-tesla.html.

Synopsys Editorial Team. (2020, July 6). *Top 10 software security best practices: Synopsys*. Software Integrity Blog. https://www.synopsys.com/blogs/software-security/top-10-software-security-best-practices/.

Wikimedia Foundation. (2021, March 24). *Bug bounty program*. Wikipedia. https://en.wikipedia.org/wiki/Bug_bounty_program.