THE CONVERSATION
Academic rigor, journalistic flair

What are software vulnerabilities, and why are there so many of them?

May 22, 2017 11.47pm EDT

It's software: There's always a way in. BeeBright via shutterstock.com

The recent WannaCry ransomware attack spread like wildfire, taking advantage of flaws in the Windows operating system to take control of hundreds of thousands of computers worldwide. But what exactly does that mean?

It can be useful to think of hackers as burglars and malicious software as their burglary tools. Having researched cybercrime and technology use among criminal populations for more than a decade, I know that both types of miscreants want to find ways into secure places – computers and networks, and homes and businesses. They have a range of options for how to get in.

Some burglars may choose to simply smash in a window or door with a crowbar, while others may be stealthier and try to pick a lock or sneak in a door that was left open. Hackers operate in a similar fashion, though they have more potential points of entry than a burglar, who is typically dependent on windows or doors.

The weaknesses hackers exploit aren't broken windowpanes or rusty hinges. Rather, they are flaws in software programs running on a computer. Programs are written by humans, and are inherently imperfect. Nobody writes software completely free of errors that create openings for potential attackers.

What are these flaws, really?

In simple terms, a vulnerability can be an error in the way that user management occurs in the system, an error in the code or a flaw in how it responds to certain requests. One common vulnerability allows an attack called a SQL injection. It works on websites that query databases, such as to search for keywords. An attacker creates a query that itself contains code in a database programming language called SQL.

If a site is not properly protected, its search function will execute the SQL commands, which can allow

Author

Thomas Holt

Associate Professor of Criminal Justice, Michigan State University

the attacker access to the database and potentially control of the website.

Similarly, many people use programs that are supported by the Java programming language, such as Adobe Flash Player and various Android applications. There are numerous vulnerabilities in the Java platform, all of which can be exploited in different ways, but most commonly through getting individuals to download “plug-ins” or “codecs” to software. These plug-ins actually contain malicious code that will take advantage of the vulnerability and compromise the machine.

Flaws are everywhere

Vulnerabilities exist in all types of software. Several versions of the Microsoft Windows operating system were open to the WannaCry attack. For instance, the popular open-source web browser Firefox has had more than 100 vulnerabilities identified in its code each year since 2009. Fifteen different vulnerabilities have been identified in Microsoft Internet Explorer browser variants since the start of 2017.

Software development is not a perfect process. Programmers often work on timelines set by management teams that attempt to set reasonable goals, though it can be a challenge to meet those deadlines. As a result, developers do their best to design secure products as they progress but may not be able to identify all flaws before an anticipated release date. Delays may be costly; many companies will release an initial version of a product and then, when they find problems (or get reports from users or researchers), fix them by releasing security updates, sometimes called patches because they cover the holes.

But software companies can’t support their products forever – to stay in business, they have to keep improving programs and selling copies of the updated versions. So after some amount of time goes by, they stop issuing patches for older programs.

Not every customer buys the latest software, though – so many users are still running old programs that might have unpatched flaws. That gives attackers a chance to find weaknesses in old software,

even if newer versions don't have the same flaws.

Exploiting the weaknesses

Once an attacker identifies a vulnerability, he can write a new computer program that uses that opportunity to get into a machine and take it over. In this respect, an exploit is similar to the way burglars use tools like crowbars, lock picks or other means of entry into a physical location.

They find a weak point in the system's defenses, perhaps a network connection that hasn't been properly secured. If attackers can manage to gain contact with a target computer, they can learn about what sort of system it is. That lets them identify particular approaches – accessing specific files or running certain programs – that can give them increasing control over the machine and its data. In recent years, attackers began targeting web browsers, which are allowed to connect to the internet and often to run small programs; they have many vulnerabilities that can be exploited. Those initial openings can give an attacker control of a target computer, which in turn can be used as a point of intrusion into a larger sensitive network.

Sometimes the vulnerabilities are discovered by the software developers themselves, or users or researchers who alert the company that a fix is needed. But other times, hackers or government spy agencies figure out how to break into systems and don't tell the company. These weaknesses are called "zero days," because the developer has had no time to fix them. As a result, the software or hardware has been compromised until a patch or fix can be created and distributed to users.

The best way users can protect themselves is to regularly install software updates, as soon as updates are available.

Hacking **Cybersecurity** **Hackers** **Malware** **Computer hacking** **Hacker** **Selected stories** **WannaCrypt** **WannaCry**