14.2.8 IoT Hacking Facts

Many IoT devices lack strong security policies, making them easy targets for hackers to attack and misuse in numerous ways. This lesson explores the IoT hacking methodology.

This lesson covers the following topics:

- IoT hacking methodology
- IoT hacking tools

IoT Hacking Methodology

The IoT hacking methodology steps are listed in the following table.

Step	Description
Information gathering	Hackers look for IP addresses, types of devices, protocols used, open ports, manufacturing company, manufacturing number, device locations, and other details that can help determine weaknesses available for exploitation. Tools such as Shodan, Censys, and Thingful can search the internet and gather information about potential targets. These tools use a process called banner grabbing. Banners are text-based information files that display a lot of system information such as OS version, services, domain and host names, organizations, countries, passwords, port numbers, and so on. These tools go around the internet interrogating ports and collecting these banners. They then display the information gathered on the screen for the hacker. An alternative to banner grabbing is to sniff network communication to detect devices. Some of the best tools for IoT sniffing are Foren6, Zniffer, CloudShark, and Wireshark.
Vulnerability scanning	BeyondTrust, a cybersecurity company, has a product named Retina IoT scanner. Retina IoT scanner gives an attacker's view of all IoT devices and their vulnerabilities across network, web, mobile, cloud, and virtual infrastructures. It uses server banner and header data to determine the make and model of each IoT device. Then Retina IoT performs a test to check whether the devices are using default or hard-coded credentials for Telnet, SSH, or basic HTTP authentication, which are the easiest to hack. Basically, the hacker uses this tool by simply specifying a target IP or an IP range and Retina IoT detects vulnerabilities that can be exploited.
Launch attacks	Once the hacker knows the devices that are available for attack and what vulnerabilities can be exploited, the next step is to launch a planned attack. Some of the most common attacks on IoT devices and systems are DDoS, rolling code, jamming signal, Sybil, Man-in-the-middle, data theft, and identity theft. There are tools that help with launching attacks. For example, the RFCrack tool makes it possible to perform rolling code attacks, replay attacks, and jamming attacks on devices. There's also a tool called KillerBee that specializes in attacking Zigbee and other IEEE 802.15.4 networks. 802.15.4 networks, including Zigbee, are low power, low data rate, and close proximity (personal area) wireless ad hoc networks.
Gain remote access	During an attack, the hacker's main goal is to remotely gain access to a device, then launch and control an attack while remaining undetected. Depending on the weaknesses found in the device, the hacker might use the device as a backdoor to gain access to a network without infecting systems protected by a firewall, antivirus software, or an IDS/IPS. Once the hacker has established remote access through an IoT device, it can be used to launch attacks on other devices within the network.
Maintain access	Once the hacker has established remote access and attacks are being launched, it becomes imperative to maintain access for as long as possible. The longer the hacker maintains control of a system, the more elaborate his attacks will become. Some ways to remain undetected are clearing the logs, updating firmware, and using tools like Trojans and backdoors. Hackers also use tools like Firmware Mod Kit, Firmalyzer Enterprise, and Firmware Analysis Toolkit to continue exploiting the firmware in the device or devices being hacked. For example, Firmware Mod Kit lets a hacker deconstruct and reconstruct firmware images for several embedded services. This kit focuses more on Linux-based routers. However, it can work with most firmware that uses common formats and file systems like TRX/ulmage and SquashFS/CramFS.

IoT Hacking Tools

The following table describes three prominent hacking tools.

Tool	Description
Censys	Censys is a public search engine and data processing company. They get their data by scanning the internet continuously. Censys can

	detect specific vulnerable devices and networks. It then creates statistical reports on broad usage patterns and trends. Censys monitors the internet non-stop in real time, then analyzes the data it discovers. Censys allows its users to see the extent to which any given network could be exploited. It also finds new threats and assesses their global impact. Another interesting thing about Censys is that it gathers data on hosts and websites by using ZMap and ZGrab to scan the IPv4 address space every day. By doing so, these programs can keep a database of host and website configurations.
Z-Wave Sniffer	Z-Wave Sniffer (Zniffer) is a hardware tool that finds smart device traffic in a network. Some of its more prominent features are real-time monitoring; packet capture from all Z-wave networks; upgradeable firmware; support for Windows, MAC OS, and Linux; and compatiblity with all Z-wave controllers like Fibaro, Homeseer, Tridium Niagara, Z-Way, SmartThings, Vera.
beSTORM	beSTORM is a smart fuzzer that finds buffer overflow weaknesses. It automates and documents the process of delivering malicious input and then watches for unpredicted responses from an application. beSTORM can test over 50 protocols and still provide automated binary and textual analysis, advanced debugging, and stack tracing. Because of its automated protocol fuzzing techniques, beSTORM is a black-box auditing tool. It’s very intelligent in its approach. First, it tries the most likely scenarios. Then it progresses through virtually every attack combination until it finds application glitches, which mean the combination attack was successful. This is a much faster approach to find security vulnerabilities than most other tools allow. beSTORM can be used with multiple processors or multiple machines to run a parallel processing audit, shortening the testing duration.