E-ISAC Private: Sector Members and Partner Organizations (TLP: White)
Recommended Audience: Public

Internet of Things DDoS White Paper

October 24, 2016

Over the past several months, existing attack surfaces and new malware payloads were exploited in unique ways, using custom attack software. The E-ISAC developed the following recommendations for defensive capabilities in the Electricity Subsector with suggestions to improve the overall posture of network security and cyber security within our community. Security, if considered at all, is typically an afterthought for devices designed to be used as part of the Internet of Things (IoT). Cyber security practitioners agree that nearly all devices on the Internet are more likely to be attacked because of the general omission of security in the design process of these new devices. Due to the highly interconnected state of the IoT, the insecurity built into systems as mundane as consumer products and toys can now be leveraged against systems as critical as industrial controls, such as those used in the electric power industry.

Recent attacks highlight the scale of network bandwidth that can be unleashed upon connected systems. A new form of attack is a class known as the Non-Reflection Distributed Denial of Service (DDoS) Attack. This new technique uses very large numbers of devices typically classified as “Things” in the terminology of the IoT, that can be harnessed from all areas of the Internet rather than a small number of networks. This massive scale of devices had successfully generated attack throughput rates on the order of one Terabit-per-second (Tbps) or more.

Most recently, on the morning of October 21, 2016, a DDoS attack against the Dyn Managed Domain Name System (DNS) infrastructure occurred in four stages, resulting in 1.2 Tbps of network throughput (also referred to as “bandwidth”) being used against the DNS address provider’s infrastructure. It is apparent that such attacks are escalating in scale, as this is the highest throughput DDoS attack seen to date.

Non-Reflection DDoS Attack

On the night of September 20, 2016, one of the largest DDoS attacks seen on the Internet up to that time took place. Consuming bandwidth in excess of 600 Gbps, it caused stress to the site’s Internet service provider (ISP). A hosting provider in France was similarly attacked by simultaneous DDoS attacks that totaled over 1 Tbps.

These attacks differed from other common attacks, such as a DNS reflection attack. In those attacks, unmanaged “open” DNS servers on the Internet are used to create huge traffic floods against target systems by forging the DNS requests so that they appear to come from the target’s network. When the DNS servers respond to the large amount of requests, they reply to the spoofed target address, flooding it

with responses. By making multiple requests for large DNS records, the attacker can create what is known as an amplified attack.

The non-reflection form of attack that was used in the more recent DDoS attacks meant that this attack was carried out using a large collection – possibly hundreds of thousands – of hacked systems. To create an attack network that large required devices that are considered to be components of the IoT. Those "Things" can be devices, such as routers, cameras, digital video recorders, video monitors, game consoles, and other Internet Protocol-enabled items with access to the Internet and protected by weak or hard-coded passwords, and default – usually well-known – user names.

Tbps DDoS Attack

Devices using high bandwidth connections, such as security cameras in plants, facilities, substations, and switchyards have the potential to create a substantive impact on the Electricity Subsector.

There are several factors highlighting the wide attack surface that similar devices provide, including:

- usually open access to the Internet;
- the use of default login credentials and weak passwords that are implemented across entire product lines;
- implementation of common operating systems without the benefit of deactivated daemons or services, and removed executable files that could be remotely or programmatically activated.

According to a report by Flashpoint and Level3 Communications, the source code for this malware (written in C and known by various names, including Bashlite) was leaked in 2015 and exists as more than a dozen variants for multiple architectures running on Linux. The malware implements a standard client/server architecture. Each botnet spreads to new hosts by scanning for vulnerable devices in order to install the malware using one of two techniques. The first technique instructs "bots" to scan for Telnet servers and attempts to "brute force" the username and password from a list of known default credentials to gain access to new devices. The other technique scans networks to find new bots to infect and use to increase the size of existing botnets.

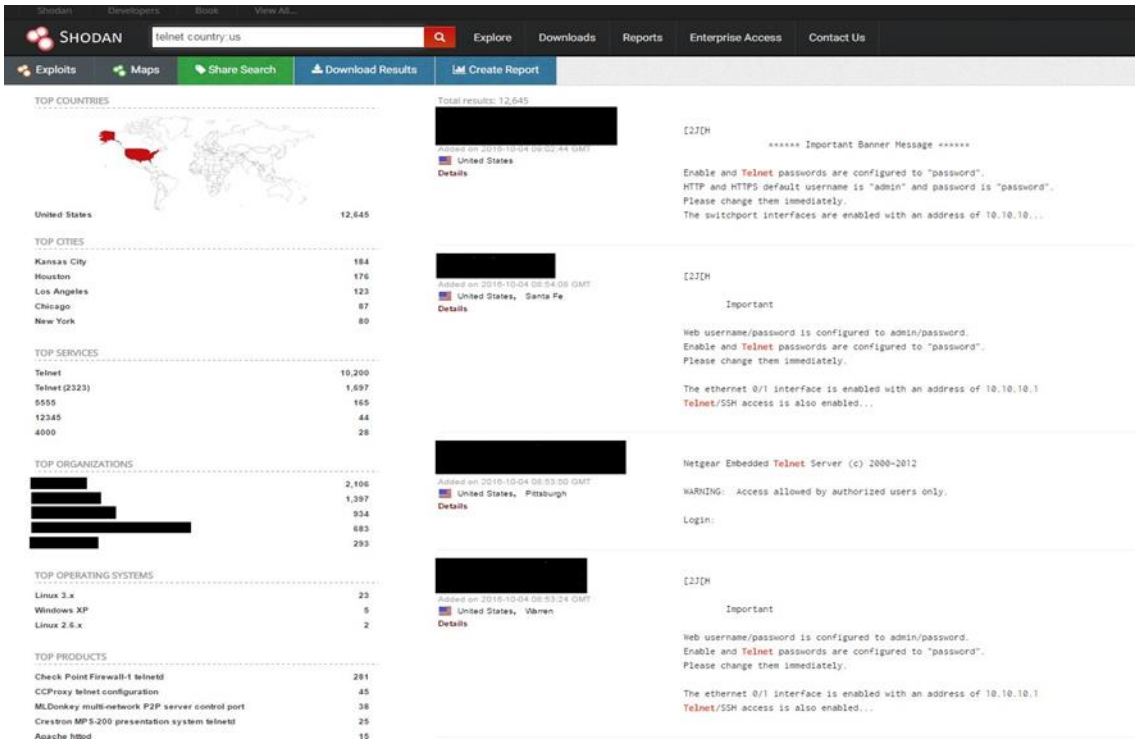
Another, more advanced malware is known as Mirai, the source code for which was posted on September 30, 2016, on the Hackforums site. Mirai exploits a version of Linux known as BusyBox, which is used in various IoT devices, including video cameras and digital video recorders (DVRs). These devices are mass-produced using default credentials, which are set at the time of production and typically not changed by users, making them easy targets. While both Mirai and Bashlight exploit the same vulnerabilities via the Telnet protocol, Mirai is more advanced. Mirai continuously scans for devices using factory default or hard-coded usernames and passwords, then uses an encrypted tunnel to communicate between the devices and command and control (C2) servers that send instructions to them. Since Mirai uses encrypted traffic, it prevents security researchers from monitoring the command and data traffic. Mirai bots can also take control of bots infected with Bashlight. Although the scale of these attacks is impressive, the techniques used to infect and attack are well known.

Shodan

Shodan is a search engine that looks for devices on the Internet. Unlike a typical search engine, it searches the access methods to those devices, rather than the content of pages or applications. Shodan is designed to search the Internet of Things.

Using a simple query, such as **port: 23,2323 net: [address range]**, Shodan will return any Internet-facing devices listening on ports 23 and 2323 in a specific address range, as shown below:

Another simple query will search for all connections related to Telnet access on systems based in the United States using the parameters **telnet country: US**, as shown below:

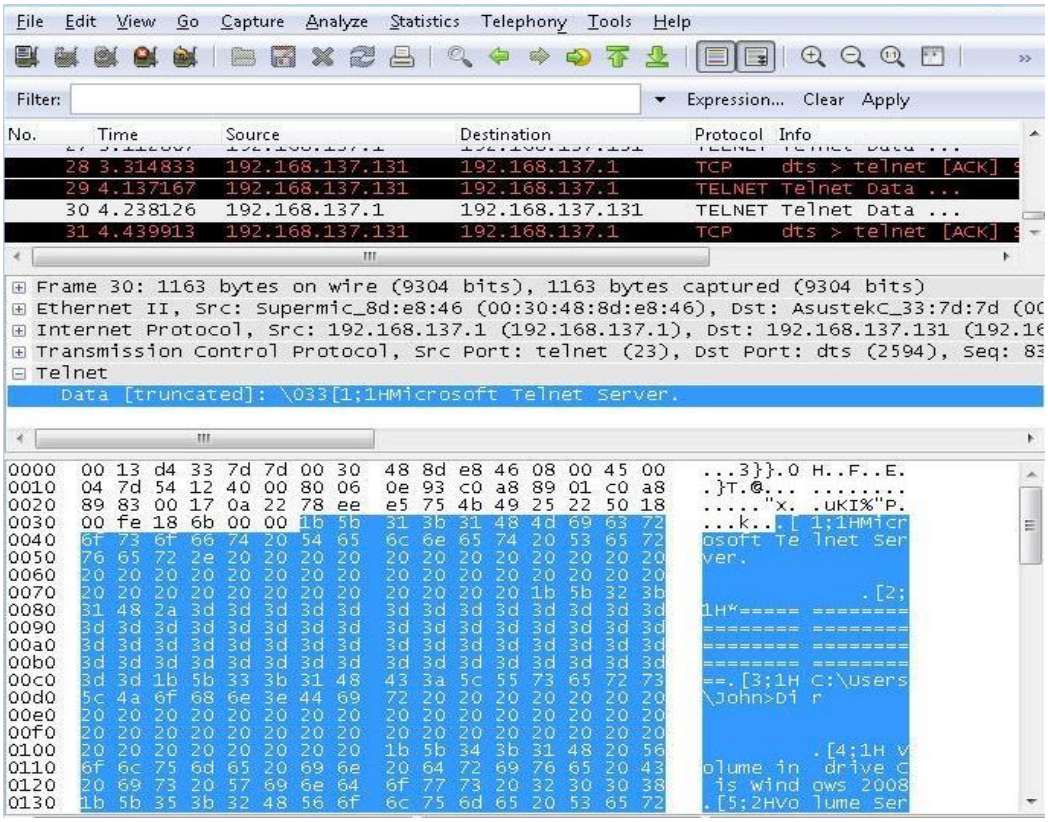


It should be noted that in this search session, several devices with the administrative login and password are displayed to systems on the global Internet. It is advised that running such a query (**telnet net: [address range]**) be performed in Shodan to determine if any such access exists in the environment. The image above is only one page showing several of thousands of devices open to the public Internet using this protocol. It also highlights the number of devices that exist on the Internet that are openly accessible and still have the manufacturer's default credentials set. This is a potentially dangerous situation, as these devices can be readily accessed via the IoT.

By clicking on the **Details** link on any entry in the returned list, Shodan will display all externally available listening services running on a device, including any banner, login, or certificate information that the service may provide. In the image above, default login credentials were also visible. Other information that can be used to provide additional reconnaissance that may be used by an attacker could be exposed using this same method.

The Danger of Using Telnet Access

The image below shows a Wireshark screen capture of a portion of a Telnet session. The commands being sent as cleartext characters as transmitted, and results of their actions are clearly seen in the information in the far right column. The graphic illustrates an interactive session as if physically attached to the system, rather than a virtual session via the Internet.



Metasploit

Metasploit is an open source intelligence and penetration testing platform that is used to find, exploit, and validate vulnerabilities.

One security site listed several Metasploit modules directed at various IoT devices:

- Network Video Camera AUTHENTICATED REMOTE COMMAND EXECUTION
- Network Video Camera AUTHENTICATED TELNET INJECTION
- PLC REMOTE START/STOP COMMAND
- Router SERVICE COMMAND INJECTION
- Network Camera FILE UPLOAD
- DVR RTSP REQUEST REMOTE CODE EXECUTION

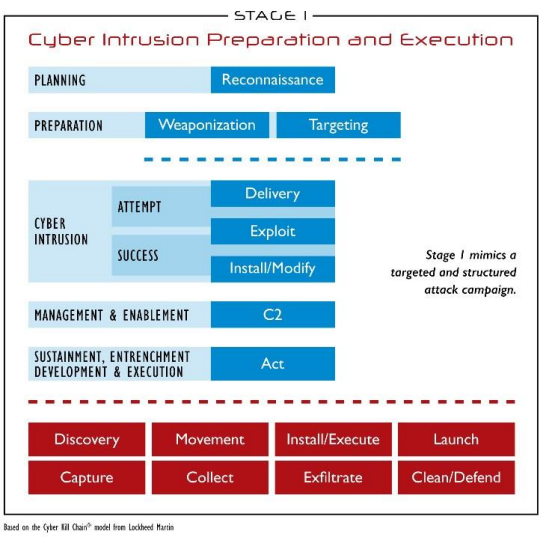
The nature of Shodan and Metasploit differ in exploring the IP address space of a network. Shodan searches are not intrusive. Shodan searches against a database; it will not actually interact with devices. Metasploit, on the other hand, will actively scan a network searching for devices with vulnerabilities that can be exploited and can interact with those devices using those vulnerabilities.

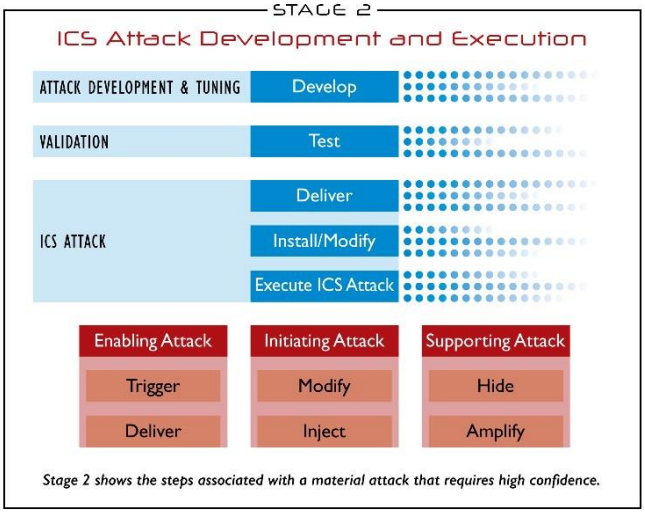
Devices that comprise networks of IT resources tend to respond better to scans by tools, such as Metasploit, versus Industrial Control System (ICS) devices that have been known to behave in unexpected ways, such as resetting, failing to respond to controllers, etc.

The issue highlighted by the exploit of a high level of connections can create devastating attacks due to the massive scales involved. This has been seen on the Internet "in the wild" in the two instances highlighted in the section on the Non-Reflection Attack. A very real concern is that the development of a large-scale botnet comprised of compromised "IoT bots" is capable of generating multiple, large-scale attacks in a tight linear or nearly simultaneous manner against multiple services. The concept of an automated series of attacks using Telnet is a very important distinction to make. While the current attacks have a primary focus on the Telnet protocol, other attacks using other protocols are possible as well.

Breaking the Kill Chain

Examining these attacks with an eye to the ICS Kill Chain, developed by the SANS Institute, and based on foundational work by researchers at Lockheed Martin, the attack mechanisms can be divided into two stages: *Intrusion Preparation and Execution,* and *Attack Development and Execution*. The *Intrusion Preparation and Execution* stage is divided into five phases: *Planning*, *Preparation*, *Cyber Intrusion*, *Management & Enablement,* and *Sustainment, Entrenchment, Development & Execution.* The *Attack Development and Execution* stage is divided into *Develop*, *Test*, *Deliver*, *Install/Modify* and *Execute*, as shown below.





The first stage of an ICS cyber attack is categorized as activities that would typically be classified as espionage or intelligence-gathering. It is in Stage 2 where the attacker uses the knowledge gained previously in Stage 1 to develop and test a set of capabilities that can be used to attack the ICS.

For example, by the time that the Mirai and Bashlight DDoS attacks were realized on the network, all of Stage 1 and the first two phases of Stage 2 in the ICS Kill Chain, *Attack Development & Tuning,* and *Validation*, had been completed. A botnet can continue to infect similarly susceptible devices that it finds on the Internet by modifying its tactics; for instance, by using a different vendor's default account and password scheme and searching for devices manufactured by that vendor based on updated commands from its C2 network. The *ICS Attack* phase's *Deliver* process is carried out by use of a file transfer mechanism, which moves the new code into position by transmitting it to the infected bots. The *Execute* phase, in this instance, acts concurrently with the *Install/Modify* phase, as infected bots are managed and used in rapid succession. They then may be redeployed to attack other victim systems simply by changing an IP address on a rudimentary management interface, for instance. The *Deliver* process is typically carried out by the use of a command and control (C2) network. Mirai, for example, has been found by security researchers to have a C2 network comprised of approximately 200 systems.

Recommendations

In light of these new DDoS attack techniques and the wide availability of tools that can identify vulnerable systems, the E-ISAC recommends that businesses and organizations examine their Internet-facing systems to ensure that:

- Internet-facing devices are inventoried and examined for vulnerabilities;
- Internet-facing devices have sufficient business justification for being publicly exposed;
- Utility-owned and managed systems that are exposed to the Internet have adequate protections in place to prevent exploit.

The E-ISAC also recommends the following technical steps to reduce an organization's attack surface and mitigate the risk of using IoT devices that might be needed for business or operational purposes:

1. Avoid permitting direct, unprotected, public Internet access to ICS devices (e.g., connecting devices to the Internet without providing access control mechanisms, such as (firewalls). This includes even seemingly innocuous devices such as security cameras, digital video recorders (DVRs), video monitors, printer, or their servers or controllers.
2. Perform a self-evaluation of your organization's Internet address space using a tool, such as Shodan, to discover what components of your infrastructure are exposed to the Internet. Register for a free account on the Shodan.io website and perform the search "net:v.w.x.y/zz" (example: net:192.168.1.0/24) to search your utility's public netblock address space. (Contact the E-ISAC for assistance with search syntax or interpretation).
3. Perform a risk assessment of the discovered Internet connected devices to determine if potential risks are acceptable.
4. Where possible, enforce changes of default login credentials, user names, and default manufacturer passwords, especially on systems that are connected to the Internet, as these are widely known.
5. Where possible, prohibit the use of "administrator" or "root" accounts on systems that are connected to the Internet, recognizing that there may be situations where a device will not operate or interoperate with the industrial control environment without a specific default administrative account. Implement the "Principle of Least Privilege" on systems that are connected to the Internet.
6. It is strongly suggested to restrict or eliminate the use of the Telnet protocol and similar protocols such as the File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Remote Login (RLOGIN), and Remote Shell (RSH) because they are inherently insecure. In all of these protocols, user credentials, passwords, commands, and data are sent in cleartext--meaning that the information can be read by even the most rudimentary network traffic monitoring tool. If specific devices allow, it is recommended to utilize the encrypted versions of these protocols such as Secure Shell (SSH) and Secure File Transfer Protocol (SFTP).
7. Avoid acquisition or implementation of systems that allow users or computers from the Internet to gain privileged access (or access administrative interfaces) on Internet-facing systems. Privileged activities can include: system shutdown, firmware update, and modification to access controls. Consider using procurement language guidance, as appropriate, when acquiring ICS or components.

Summary

Integrating the toolset of locating vulnerabilities in an environment with a search tool, such as Shodan, with the mitigation of attack tools, such as exploits developed in Metasploit, is a two-stage posture to defend against advanced attacks. There are two foci for the future of high-capacity attacks: Compromise at Scale and Delivery of Malware at Scale. These are game-changing threats in terms of cyber security, as this has the potential to use the immense scale of the IoT against a victim or multiple victims with vulnerabilities at high throughput rates. They highlight the possibility of two potential forms of attack: high volume-DDoS or varying volumes of custom payload attacks – even multiple forms of attack in each distributed botnet. These attacks must be mitigated to protect the systems that comprise critical infrastructure.