TTPs of the World’s Most Dangerous Attackers

Robert Lipovsky | Senior Malware Researcher
APT case study #1
Locate  Lock  Delete  Recover
Built into the BIOS or firmware during the manufacturing process of most major device manufacturers, we are able to provide our customers with the only security solution that can withstand a factory reset, installation of a new OS, or even a complete hard drive replacement.
LoJack Architecture

1. UEFI/BIOS module executes
   - Contains persistent agent and its dropper
   - Replaces legitimate autochk.exe

2. Windows early boot
   - autochk.exe
   - Drops rpcnetp.exe - small agent
   - Installs it as a service
LoJack Architecture

1. UEFI/BIOS module:
   - Contains persistent agent and its dropper
   - Replaces legitimate autochk.exe

2. Windows early boot:
   - autochk.exe
     - Drops rpcnetp.exe - small agent
     - Installs it as a service

3. Windows OS running:
   - rpcnetp.exe - small agent
     - Injects its DLL into svchost.exe, then Internet Explorer
     - Communicates with C&C server to download and install full recovery agent
LoJack Architecture

1. UEFI/BIOS module executes
   - UEFI/BIOS module
     - Contains persistent agent and its dropper
     - Replaces legitimate autochk.exe

2. Windows early boot
   - autochk.exe
     - Drops rpcnetp.exe - small agent
     - Installs it as a service

3. Windows OS running
   - rpcnetp.exe - small agent
     - Injects its DLL into svchost.exe, then Internet Explorer
     - Communicates with C&C server to download and install full recovery agent

4. Normal operation
   - Full recovery agent is running on the machine
Configuration file vulnerability
LoJack Architecture

1. UEFI/BIOS module executes
   - UEFI/BIOS module
   - Contains stealth agent developed
   - Replaces legitimate autochk.exe

2. Windows early boot
   - autochk.exe
   - Drops rpcnetp.exe
   - small agent
   - Installs it as a service

3. Windows OS running
   - rpcnetp.exe - small agent
   - server to download and install full recovery agent

4. Normal operation
   - Full recovery agent running on the machine

Win32/Lojox
THE HUNT

Several military and diplomatic organizations in the Balkans, Central and Eastern Europe hit
- Installs NTFS driver
- Writes Win32/Lojix to system partition
- Patch Windows Registry

A - info_efe.exe
B - ReWriter_read.exe
C - ReWriter_Binary.exe
D - EFI/Lojix
SEDNIT
a.k.a.
APT28, Fancy Bear,
Sofacy, STRONTIUM, ...
Russia’s Elite Hackers Have a Clever New Trick That's Very Hard to Fix

Midterm Election Hacking -- Who Is Fancy Bear?

How the Russians hacked the DNC and passed its emails to WikiLeaks

The Washington Post

Aug 23, 2018

Sep 27, 2018

WIRED

Forbes

July 13, 2018
LOJAX
First UEFI rootkit found in the wild, courtesy of the Sednit group
LoJax

LoJax is a UEFI rootkit used by APT28 to persist remote access software on targeted systems.\[1\]

<table>
<thead>
<tr>
<th>Domain</th>
<th>ID</th>
<th>Name</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise</td>
<td>T1112</td>
<td>Modify Registry</td>
<td>LoJax has modified the Registry key <code>\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\BootExecute</code> from <code>autocheck autochk</code> to <code>autocheck autoche</code>. [1]</td>
</tr>
<tr>
<td>Enterprise</td>
<td>T1096</td>
<td>NTFS File Attributes</td>
<td>LoJax has loaded an embedded NTFS DEX driver to be able to access and write to NTFS partitions. [1]</td>
</tr>
<tr>
<td>Enterprise</td>
<td>T1060</td>
<td>Registry Run Keys / Startup Folder</td>
<td>LoJax has modified the Registry key <code>\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\BootExecute</code> from <code>autocheck autochk</code> to <code>autocheck autoche</code> in order to execute its payload during Windows startup. [1]</td>
</tr>
<tr>
<td>Enterprise</td>
<td>T1014</td>
<td>Rootkit</td>
<td>LoJax is a UEFI BIOS rootkit deployed to persist remote access software on some targeted systems. [1]</td>
</tr>
<tr>
<td>Enterprise</td>
<td>T1019</td>
<td>System Firmware</td>
<td>LoJax is a UEFI BIOS rootkit deployed to persist remote access software on some targeted systems. [1]</td>
</tr>
</tbody>
</table>

Groups That Use This Software
System Firmware

The BIOS (Basic Input/Output System) and The Unified Extensible Firmware Interface (UEFI) or Extensible Firmware Interface (EFI) are examples of system firmware that operate as the software interface between the operating system and hardware of a computer. [1] [2] [3]

System firmware like BIOS and (U)EFI underly the functionality of a computer and may be modified by an adversary to perform or assist in malicious activity. Capabilities exist to overwrite the system firmware, which may give sophisticated adversaries a means to install malicious firmware updates as a means of persistence on a system that may be difficult to detect.

Procedure Examples

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hacking Team UEFI Rootkit</td>
<td>Hacking Team UEFI Rootkit is a UEFI BIOS rootkit developed by the company Hacking Team to persist remote access software on some targeted systems. [4]</td>
</tr>
<tr>
<td>LoJax</td>
<td>LoJax is a UEFI BIOS rootkit deployed to persist remote access software on some targeted systems. [6]</td>
</tr>
<tr>
<td>Trojan.Mebromi</td>
<td>Trojan.Mebromi performs BIOS modification and can download and execute a file as well as protect itself from removal. [8]</td>
</tr>
</tbody>
</table>

Mitigations
APT case study #2
Sandworm Team

Sandworm Team is a Russian cyber espionage group that has operated since approximately 2009. The group likely consists of Russian pro-hacktivists. Sandworm Team targets mainly Ukrainian entities associated with energy, industrial control systems, SCADA, government, and media. Sandworm Team has been linked to the Ukrainian energy sector attack in late 2015. [1] [2]

Associated Group Descriptions

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quedagh</td>
<td>Based on similarities between TTPs, malware, and targeting, Sandworm Team and Quedagh appear to refer to the same group. [1] [3]</td>
</tr>
<tr>
<td>VOODOO BEAR</td>
<td>[2]</td>
</tr>
</tbody>
</table>

Software

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>References</th>
<th>Techniques</th>
</tr>
</thead>
</table>

References

2. [Sandworm Team](https://attack.mitre.org/groups/G0034) [2]
BlackEnergy attack causing a blackout in Ukraine
DEC 2015
Підстанції

ПС Соломинка 35/10кВ
ПС Вербина 35/10кВ
ПС Дземель 35/10кВ
ПС Мазогуров 35/10кВ
ПС Чернигівська 11/0,4 кВ
Exaramel vs Industroyer

DWORD _stdcall cmd_thread(thread_param *param)
{
    // [COLLAPSED LOCAL DECLARATIONS, PRESS KEYPAD CTRL-"+" TO EXPAND]
    result1 = Ox16;
    v2 = init CMD struct(param->xml, &CMD);
    SetEvent((HANDLE)param->event);
    if ( v2 )
    {
        return 1;
    }
    cmd_struct1 = CMD;
    switch ( CMD->cmd_id )
    {
        case 1:
            result = cmd_create_process(CMD);
            goto end;
        case 2:
            result = cmd_create_process_as_user(CMD);
            goto end;
        case 3:
            result = cmd_write_file(CMD);
            goto end;
        case 4:
            result = cmd_copy_file_aka_upload(CMD);
            goto end;
        case 5:
            result = cmd_execute_shell(cmd);
            goto end;
        case 6:
            result = cmd_execute_shell_cmd_as_user(CMD);
            goto end;
        case 7:
            result = cmd_eval_UBS_code(CMD);
            break;
    }
    return result1 - result;
    break;
    default:
    break;
    }
PathCombineW acquired, ((LPCTSTR)cmd_struct1->storage_path, L"\done");
    file_write(&acquired, 0, 0);
    mem_free((LPVOID)cmd_struct1->file_content);
    mem_free((LPVOID)cmd_struct1->cmd_content);
    mem_free((LPVOID)cmd_struct1->cmd_struct);
    return result;
}

int _cdecl run_command(cmd_internal *CMD)
{
    int result; // eax
    result = LOBYTE(CMD->cmd_id) - 1;
    switch ( LOBYTE(CMD->cmd_id) )
    {
        case 1u:
            result = cmd_create_process(CMD);
            break;
        case 2u:
            result = cmd_create_process_as_user(CMD);
            break;
        case 3u:
            result = cmd_write_file(CMD);
            break;
        case 4u:
            result = cmd_copy_file_aka_upload(CMD);
            break;
        case 5u:
            result = cmd_execute_shell(CMD);
            break;
        case 6u:
            result = cmd_execute_shell_cmd_as_user(CMD);
            break;
        case 7u:
            ExitProcess(0);
            return result;
        case 8u:
            result = cmd_stop_service(CMD);
            break;
        case 9u:
            result = cmd_stop_service_as_user(CMD);
            break;
        case 0x8u:
            result = cmd_start_service_as_user(CMD);
            break;
        case 0x9u:
            result = cmd_service_change_path_to_binary_as_user(CMD);
            break;
        default:
            return result;
        case 0x0u:
        return result;
    }
Spearphishing Attachment

Spearphishing attachment is a specific variant of spearphishing. Spearphishing attachment is different from other forms of spearphishing in that it employs the use of malware attached to an email. All forms of spearphishing are electronically delivered social engineering targeted at a specific individual, company, or industry. In this scenario, adversaries attach a file to the spearphishing email and usually rely upon User Execution to gain execution.

There are many options for the attachment such as Microsoft Office documents, executables, PDFs, or archived files. Upon opening the attachment (and potentially clicking past protections), the adversary’s payload exploits a vulnerability or directly executes on the user’s system. The text of the spearphishing email usually tries to give a plausible reason why the file should be opened, and may explain how to bypass system protections in order to do so. The email may also contain instructions on how to decrypt an attachment, such as a zip file password, in order to evade email boundary defenses. Adversaries frequently manipulate file extensions and icons in order to make attached executables appear to be document files, or files exploiting one application appear to be a file for a different one.

Procedure Examples

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APT12</td>
<td>APT12 has sent emails with malicious Microsoft Office documents and PDFs attached. [88] [89]</td>
</tr>
<tr>
<td>APT19</td>
<td>APT19 sent spearphishing emails with malicious attachments in RTF and XLSM formats to deliver initial exploits. [62]</td>
</tr>
<tr>
<td>APT28</td>
<td>APT28 sent spearphishing emails containing malicious Microsoft Office attachments. [22] [23] [24] [25] [26]</td>
</tr>
<tr>
<td>APT39</td>
<td>APT39 has used spearphishing emails with an attachment to deliver files with exploits to initial victims. [25] [24]</td>
</tr>
</tbody>
</table>
Exploitation for Client Execution

Vulnerabilities can exist in software due to insecure coding practices that can lead to unanticipated behavior. Adversaries can take advantage of certain vulnerabilities through targeted exploitation for the purpose of arbitrary code execution. Oftentimes, the most valuable exploits to an offensive toolkit are those that can be used to obtain code execution on a remote system because they can be used to gain access to that system. Users will expect to see files related to the applications they commonly use to do work, so they are a useful target for exploit research and development because of their high utility.

Several types exist:

**Browser-based Exploitation**

Web browsers are a common target through Drive-by Compromise and Spear phishing Link. Endpoint systems may be compromised through normal web browsing or from certain users being targeted by links in spear phishing emails to adversary controlled sites used to exploit the web browser. These often do not require an action by the user for the exploit to be executed.

**Office Applications**

Common office and productivity applications such as Microsoft Office are also targeted through Spear phishing Attachment, Spear phishing Link, and Spear phishing via Service. Malicious files will be transmitted directly as attachments or through links to download them. These require the user to open the document or file for the exploit to run.

**Common Third-party Applications**

Other applications that are commonly seen or are part of the software deployed in a target network may also be targets.
Supply Chain Compromise

Supply chain compromise is the manipulation of products or product delivery mechanisms prior to receipt by a final consumer for the purpose of data or system compromise.

Supply chain compromise can take place at any stage of the supply chain including:

- Manipulation of development tools
- Manipulation of a development environment
- Manipulation of source code repositories (public or private)
- Manipulation of source code in open-source dependencies
- Manipulation of software update/distribution mechanisms
- Compromised/injected system images (multiple cases of removable media infected at the factory)
- Replacement of legitimate software with modified versions
- Sales of modified/counterfeit products to legitimate distributors
- Shipment interdiction

While supply chain compromise can impact any component of hardware or software, attackers looking to gain execution have often focused on malicious additions to legitimate software in software distribution or update channels. [1] [3] Targeting may be specific to a desired victim set [4] or malicious software may be distributed to a broad set of consumers but only move on to additional tactics on specific victims. [1] [8] Popular open source projects that are used as dependencies in many applications may also be targeted as a means to add malicious code to users of the dependency. [5]

Procedure Examples
Telebots supply chain attacks

TeleBots ransomware
Win32/Filecoder.NKH

MARCH 2017

Win32/Diskcoder.C aka Petya

MAY 2017

Win32/Filecoder.AESNI.C aka XData

JUNE 2017
Exploit Public-Facing Application

The use of software, data, or commands to take advantage of a weakness in an Internet-facing computer system or program in order to cause unintended or unanticipated behavior. The weakness in the system can be a bug, a glitch, or a design vulnerability. These applications are often websites, but can include databases (like SQL)\(^\text{1}\), standard services (like SMB\(^\text{2}\) or SSH), and any other applications with Internet accessible open sockets, such as web servers and related services,\(^\text{3}\) Depending on the flaw being exploited this may include Exploitation for Defense Evasion.

For websites and databases, the OWASP top 10 and CWE top 25 highlight the most common web-based vulnerabilities.\(^\text{4}\)\(^\text{5}\)

Procedure Examples

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axiom</td>
<td>Axiom has been observed using SQL injection to gain access to systems.(^\text{9})(^\text{10})</td>
</tr>
<tr>
<td>Havij</td>
<td>Havij is used to automate SQL injection.(^\text{4})</td>
</tr>
<tr>
<td>Night Dragon</td>
<td>Night Dragon has performed SQL injection attacks of extranet web servers to gain access.(^\text{8})</td>
</tr>
<tr>
<td>Soft Cell</td>
<td>Soft Cell exploited a publicly-facing server to gain access to the network.(^\text{11})</td>
</tr>
<tr>
<td>sqlmap</td>
<td>sqlmap can be used to automate exploitation of SQL injection vulnerabilities.(^\text{7})</td>
</tr>
</tbody>
</table>

Mitigations
Other interesting techniques

- TeamViewer for remote unattended access
- Parasitic infector
- Execution via HMI
ICS Alert (ICS-ALERT-14-281-01E)

Ongoing Sophisticated Malware Campaign Compromising ICS (Update E)

Original release date: December 10, 2014 | Last revised: December 09, 2016
• Modules signed by certificate stolen from industrial manufacturer
More GreyEnergy Techniques

• Selective module use; module ‘security’
• Internal C&C proxies
• C&C servers Tor relays
Impact
ATT&CK for Industrial Control Systems

ATT&CK™ for ICS is a knowledge base useful for describing the actions an adversary may take while operating within an ICS network. The knowledge base can be used to better characterize and describe post-compromise adversary behavior. Please see the overview page for more information about ATT&CK for ICS.

You may start with the following links to become more familiar with ATT&CK for ICS:

- Full list of ATT&CK for ICS techniques
- Software used by ICS threats
- Adversary groups from ICS related incidents
- Assets present in ICS
- Contribute or contact us

The MITRE ATT&CK for ICS Matrix™ is an overview of the tactics and techniques described in the ATT&CK for ICS knowledge base. It visually aligns individual techniques under the tactics in which they can be applied. Some techniques span more than one tactic because they can be used for different purposes.
Software: Industroyer, CRASHOVERRIDE

**Industroyer** is a sophisticated piece of malware designed to cause an [Impact](#) to the working processes of Industrial Control Systems (ICS), specifically ICSs used in electrical substations. [1] Industroyer was alleged to be used in the attacks on the Ukrainian power grid in December 2016. [2][3][4][5]

### Contents [hide]

1. Associated Software Descriptions
2. Techniques Used
3. Groups
4. References

### Associated Software Descriptions

- Industroyer - [1]
- CRASHOVERRIDE - [2][5]

### Techniques Used

- **Data Historian Compromise** - In Industroyer, after pivoting into the ICS environment, the adversary gained [Initial Access](#) to devices involved with critical process operations through a Microsoft Windows Server 2003 running a SQL Server. [4]

- **Block Command Message** - In Industroyer the first COM port from the configuration file is used for the actual communication and the two other COM ports are just opened to prevent other processes accessing them. Thus, the IEC 101 payload component is able to take over and maintain control of the RTU device. [1]

- **Block Serial COM** - In Industroyer the first COM port from the configuration file is used for the actual communication and the two other COM ports are just opened to
Industroyer impact: ICS PROTOCOL PAYLOADS
Industroyer impact: ICS PROTOCOL PAYLOADS
Industroyer impact: DENIAL OF SERVICE
Industroyer impact: DATA WIPER
Data Destruction

Adversaries may destroy data and files on specific systems or in large numbers on a network to interrupt availability to systems, services, and network resources. Data destruction is likely to render stored data irrecoverable by forensic techniques through overwriting files or data on local and remote drives.[1][2][3][4][5][6]

Common operating system file deletion commands such as `rm` and `srm` often only remove pointers to files without wiping the contents of the files themselves, making the files recoverable by proper forensic methodology. This behavior is distinct from Disk Content Wipe and Disk Structure Wipe because individual files are destroyed rather than sections of a storage disk or the disk's logical structure.

Adversaries may attempt to overwrite files and directories with randomly generated data to make it irrecoverable.[4][6] In some cases politically oriented image files have been used to overwrite data.[2][3][4][4]

To maximize impact on the target organization in operations where network-wide availability interruption is the goal, malware designed for destroying data may have worm-like features to propagate across a network by leveraging additional techniques like Valid Accounts, Credential Dumping, and Windows Admin Shares.[1][2][3][4][4]

Procedure Examples

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APT38</td>
<td>APT38 has used a custom secure delete function to make deleted files unrecoverable. [13]</td>
</tr>
<tr>
<td><strong>BlackEnergy</strong></td>
<td><strong>BlackEnergy</strong> 2 contains a “Destroy” plug-in that destroys data stored on victim hard drives by overwriting file contents. [8]</td>
</tr>
<tr>
<td>Kazuar</td>
<td>Kazuar can overwrite files with random data before deleting them. [9]</td>
</tr>
<tr>
<td>Lazarus Group</td>
<td>Lazarus Group has used a custom secure delete function to overwrite file contents with data from heap memory. [14]</td>
</tr>
</tbody>
</table>
Data Encrypted for Impact

Adversaries may encrypt data on target systems or on large numbers of systems in a network to interrupt availability to system and network resources. They can attempt to render stored data inaccessible by encrypting files or data on local and remote drives and withholding access to a decryption key. This may be done in order to extract monetary compensation from a victim in exchange for decryption or a decryption key (ransomware) or to render data permanently inaccessible in cases where the key is not saved or transmitted. In the case of ransomware, it is typical that common user files like Office documents, PDFs, images, videos, audio, text, and source code files will be encrypted. In some cases, adversaries may encrypt critical system files, disk partitions, and the MBR.

To maximize impact on the target organization, malware designed for encrypting data may have worm-like features to propagate across a network by leveraging other attack techniques like Valid Accounts, Credential Dumping, and Windows Admin Shares.

### Procedure Examples

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APT38</td>
<td>APT38 has used Hermes ransomware to encrypt files with AES256.</td>
</tr>
<tr>
<td>Jcry</td>
<td>Jcry has encrypted files and demanded Bitcoin to decrypt those files.</td>
</tr>
<tr>
<td>LockerGoga</td>
<td>LockerGoga has encrypted files, including core Windows OS files, using RSA-OAEP MGF1 and then demanded Bitcoin be paid for the decryption key.</td>
</tr>
<tr>
<td>NotPetya</td>
<td>NotPetya encrypts user files and disk structures like the MBR with 2048-bit RSA.</td>
</tr>
<tr>
<td>SamSam</td>
<td>SamSam encrypts victim files using RSA-2048 encryption and demands a ransom be paid in Bitcoin to decrypt those files.</td>
</tr>
</tbody>
</table>
WE ARE FSOCIETY
JOIN US
Thank You!

@Robert_Lipovsky