CrashOverride Malware

Systems Affected

Industrial Control Systems

Description

Technical Analysis

CrashOverride malware represents a scalable, capable platform. The modules and capabilities publically reported appear to focus on organizations using ICS protocols IEC101, IEC104, and IEC61850, which are more commonly used outside the United States in electric power control systems. The platform fundamentally abuses the functionality of a targeted ICS system's legitimate control system to achieve its intended effect. While the known capabilities do not appear to be U.S.-focused, it is important to recognize that the general TTPs used in CrashOverride could be leveraged with modified technical implementations to affect U.S.-based critical infrastructure. With further modification, CrashOverride or similar malware could have implications beyond electric power so all critical infrastructure organizations should be evaluating their systems to susceptibilities in the TTPs outlined. The malware has several reported capabilities:

1. Issues valid commands directly to remote terminal units (RTUs) over ICS protocols. As reported by Dragos, one such command sequence toggles circuit breakers in a rapid open-close-open-close pattern. This could create conditions where individual utilities may island from infected parties, potentially resulting in a degradation of grid reliability.
2. Denies service to local serial COM ports on windows devices, therefore preventing legitimate communications with field equipment over serial from the affected device.
3. Scans and maps ICS environment using a variety of protocols, including Open Platform Communications (OPC). This significantly improves the payload's probability of success.
4. Could exploit Siemens relay denial-of-service (DoS) vulnerability, leading to a shutdown of the relay. In this instance, the relay would need to be manually reset to restore functionality.
5. Includes a wiper module in the platform that renders windows systems inert, requiring a rebuild or backup restoration.

Detection

As CrashOverride is a second stage malware capability and has the ability to operate independent of initial C2, traditional methods of detection may not be sufficient to detect infections prior to the malware executing. As a result, organizations are encouraged to implement behavioral analysis techniques to attempt to identify precursor activity to

CrashOverride. As additional information becomes available on stage one infection vectors and TTPs, this alert will be updated.

NCCIC is providing a compilation of IOCs (see links above) from a variety of sources to aid in the detection of this malware. The sources provided do not constitute an exhaustive list and the U.S. Government does not endorse or support any particular product or vendor's information referenced in this report. However, NCCIC has included this data to ensure wide distribution of the most comprehensive information available and will provide updates as warranted.

Signatures

```
import “pe”
import “hash”

rule dragos_crashoverride_exporting_dlls
{
meta:
description = “CRASHOVERRIDE v1 Suspicious Export”
author = “Dragos Inc”
condition:
pe.exports(“Crash”) & pe.characteristics
}

rule dragos_crashoverride_suspcious
{
meta:
description = “CRASHOVERRIDE v1 Wiper”
author = “Dragos Inc”
strings:
$s0 = “SYS_BASCON.COM” fullword nocase wide
$s1 = “.pcmp” fullword nocase wide
$s2 = “.pcmi” fullword nocase wide
$s3 = “.pcmt” fullword nocase wide
$s4 = “.cin” fullword nocase wide
condition:
pe.exports(“Crash”) and any of ($s*)
}

rule dragos_crashoverride_name_search {
meta:
description = “CRASHOVERRIDE v1 Suspicious Strings and Export”
author = “Dragos Inc”
strings:
$s0 = “101.dll” fullword nocase wide
$s1 = “Crash101.dll” fullword nocase wide
$s2 = “104.dll” fullword nocase wide
$s3 = “Crash104.dll” fullword nocase wide
$s4 = “61850.dll” fullword nocase wide
```

```
$s5 = “Crash61850.dll” fullword nocase wide
$s6 = “OPCClientDemo.dll” fullword nocase wide
$s7 = “OPC” fullword nocase wide
$s8 = “CrashOPCClientDemo.dll” fullword nocase wide
$s9 = “D2MultiCommService.exe” fullword nocase wide
$s10 = “CrashD2MultiCommService.exe” fullword nocase wide
$s11 = “61850.exe” fullword nocase wide
$s12 = “OPC.exe” fullword nocase wide
$s13 = “haslo.exe” fullword nocase wide
$s14 = “haslo.dat” fullword nocase wide
condition:
any of ($s*) and pe.exports(“Crash”)
}

rule dragos_crashoverride_hashes {
meta:
description = “CRASHOVERRIDE Malware Hashes”
author = “Dragos Inc”

condition:
filesize < 1MB and
hash.sha1(0, filesize) == “f6c21f8189ced6ae150f9ef2e82a3a57843b587d” or
hash.sha1(0, filesize) == “cccce62996d578b984984426a024d9b250237533” or
hash.sha1(0, filesize) == “8e39eca1e48240c01ee570631ae8f0c9a9637187” or
hash.sha1(0, filesize) == “2cb8230281b86fa944d3043ae906016c8b5984d9” or
hash.sha1(0, filesize) == “79ca89711cdaedb16b0ccccfdcfbd6aa7e57120a” or
hash.sha1(0, filesize) == “94488f214b165512d2fc0438a581f5c9e3bd4d4c” or
hash.sha1(0, filesize) == “5a5fafbc3fec8d36fd57b075ebf34119ba3bff04” or
hash.sha1(0, filesize) == “b92149f046f00bb69de329b8457d32c24726ee00” or
hash.sha1(0, filesize) == “b335163e6eb854df5e08e85026b2c3518891eda8”
}

rule dragos_crashoverride_moduleStrings {
meta:
description = “IEC-104 Interaction Module Program Strings”
author = “Dragos Inc”
strings:
$s1 = “IEC-104 client: ip=%s; port=%s; ASDU=%u” nocase wide ascii
$s2 = “ MSTR ->> SLV” nocase wide ascii
$s3 = “ MSTR <<- SLV” nocase wide ascii
$s4 = “Unknown APDU format !!!” nocase wide ascii
$s5 = “iec104.log” nocase wide ascii
condition:
any of ($s*)
}

rule dragos_crashoverride_configReader
{
```

```
meta:
description = “CRASHOVERRIDE v1 Config File Parsing”
author = “Dragos Inc”
strings:
$s0 = { 68 e8 ?? ?? ?? 6a 00 e8 a3 ?? ?? ?? 8b f8 83 c4 ?8 }
$s1 = { 8a 10 3a 11 75 ?? 84 d2 74 12 }
$s2 = { 33 c0 eb ?? 1b c0 83 c8 ?? }
$s3 = { 85 c0 75 ?? 8d 95 ?? ?? ?? ?? 8b cf ?? ?? }
condition:
all of them
}


rule dragos_crashoverride_configReader
{
meta:
description = “CRASHOVERRIDE v1 Config File Parsing”
author = “Dragos Inc”
strings:
$s0 = { 68 e8 ?? ?? ?? 6a 00 e8 a3 ?? ?? ?? 8b f8 83 c4 ?8 }
$s1 = { 8a 10 3a 11 75 ?? 84 d2 74 12 }
$s2 = { 33 c0 eb ?? 1b c0 83 c8 ?? }
$s3 = { 85 c0 75 ?? 8d 95 ?? ?? ?? ?? 8b cf ?? ?? }
condition:
all of them
}

rule dragos_crashoverride_weirdMutex
{
meta:
description = “Blank mutex creation assoicated with CRASHOVERRIDE”
author = “Dragos Inc”
strings:
$s1 = { 81 ec 08 02 00 00 57 33 ff 57 57 57 ff 15 ?? ?? 40 00 a3 ?? ?? ?? 00 85 c0 }
$s2 = { 8d 85 ?? ?? ?? ff 50 57 57 6a 2e 57 ff 15 ?? ?? ?? 00 68 ?? ?? 40 00}
condition:
all of them
}

rule dragos_crashoverride_serviceStomper
{
meta:
description = “Identify service hollowing and persistence setting”
author = “Dragos Inc”
strings:
$s0 = { 33 c9 51 51 51 51 51 51 ?? ?? ?? }
$s1 = { 6a ff 6a ff 6a ff 50 ff 15 24 ?? 40 00 ff ?? ?? ff 15 20 ?? 40 00 }
condition:
```

```
all of them
}

rule dragos_crashoverride_wiperModuleRegistry
{
meta:
description = “Registry Wiper functionality assoicated with CRASHOVERRIDE”
author = “Dragos Inc”
strings:
$s0 = { 8d 85 a0 ?? ?? ?? 46 50 8d 85 a0 ?? ?? ?? 68 68 0d ?? ?? 50 }
$s1 = { 6a 02 68 78 0b ?? ?? 6a 02 50 68 b4 0d ?? ?? ff b5 98 ?? ?? ?? ff 15
04 ?? ?? ?? }
$s2 = { 68 00 02 00 00 8d 85 a0 ?? ?? ?? 50 56 ff b5 9c ?? ?? ?? ff 15 00 ?? ?? ?? 85
c0 }
condition:
all of them
}

rule dragos_crashoverride_wiperFileManipulation
{
meta:
description = “File manipulation actions associated with CRASHOVERRIDE wip¬er”
author = “Dragos Inc”
strings:
$s0 = { 6a 00 68 80 00 00 00 6a 03 6a 00 6a 02 8b f9 68 00 00 00 40 57 ff 15
1c ?? ?? ?? 8b d8 }
$s2 = { 6a 00 50 57 56 53 ff 15 4c ?? ?? ?? 56 }
condition:
all of them
}
```

Impact

A successful network intrusion can have severe impacts, particularly if the compromise becomes public and sensitive information is exposed. Possible impacts include:

- temporary or permanent loss of sensitive or proprietary information,
- disruption to regular operations,
- financial losses incurred to restore systems and files, and
- potential harm to an organization’s reputation.

Solution

Properly implemented defensive techniques and common cyber hygiene practices increase the complexity of barriers that adversaries must overcome to gain unauthorized access to critical information networks and systems. In addition, detection and prevention mechanisms can expose malicious network activity, enabling

organizations to contain and respond to intrusions more rapidly. There is no set of defensive techniques or programs that will completely avert all attacks however, layered cybersecurity defenses will aid in reducing an organization's attack surface and will increase the likelihood of detection. This layered mitigation approach is known as defense-in-depth.
NCCIC has based its mitigations and recommendations on its analysis of the public reporting of this malware and will be provide updates as more information becomes available.
Critical infrastructure companies should ensure that they are following best practices, which are outlined in the Seven Steps to Effectively Defend Industrial Control Systems document produced jointly by DHS, NSA, and FBI.

Application Whitelisting

Application whitelisting (AWL) can detect and prevent attempted execution of malware uploaded by adversaries. Application whitelisting hardens operating systems and prevents the execution of unauthorized software. The static nature of some systems, such as database servers and human-machine interface (HMI) computers make these ideal candidates to run AWL. NCCIC encourages operators to work with their vendors to baseline and calibrate AWL deployments.
Operators may choose to implement directory whitelisting rather than trying to list every possible permutation of applications in an environment. Operators may implement application or application directory whitelisting through Microsoft Software Restriction Policy (SRP), AppLocker, or similar application whitelisting software. Safe defaults allow applications to run from PROGRAMFILES, PROGRAMFILES(X86), SYSTEM32, and any ICS software folders. All other locations should be disallowed unless an exception is granted.

Manage Authentication and Authorization

This malware exploits the lack of authentication and authorization in common ICS protocols to issue unauthorized commands to field devices. Asset owners/operators should implement authentication and authorization protocols to ensure field devices verify the authenticity of commands before they are actioned. In some instances, legacy hardware may not be capable of implementing these protections. In these cases, asset owners can either leverage ICS firewalls to do stateful inspection and authentication of commands, or upgrade their control field devices.

Adversaries are increasingly focused on gaining control of legitimate credentials, especially those associated with highly privileged accounts. Compromising these credentials allows adversaries to masquerade as legitimate users, leaving less evidence of compromise than more traditional attack options (i.e., exploiting vulnerabilities or uploading malware). For this reason, operators should implement multi-factor authentication where possible and reduce privileges to only those needed for a user's duties. If passwords are necessary, operators should implement secure password policies, stressing length over complexity. For all accounts, including system and non-interactive accounts, operators should ensure credentials are unique, and changed, at a minimum, every 90 days.

NCCIC also recommends that operators require separate credentials for corporate and control network zones and store them in separate trust stores. Operators should never share Active Directory, RSA ACE servers, or other trust stores between corporate and control networks. Specifically, operators should:

- Decrease a threat actor's ability to access key network resources by implementing the principle of least privilege;
- Limit the ability of a local administrator account to login from a local interactive session (e.g., "Deny access to this computer from the network") and prevent access via a remote desktop protocol session;
- Remove unnecessary accounts, groups, and restrict root access;
- Control and limit local administration; and
- Make use of the Protected Users Active Directory group in Windows Domains to further secure privileged user accounts against pass-the-hash attacks.

Handling Destructive Malware

Destructive malware continues to be a threat to both critical infrastructure and business systems. NCCIC encourages organizations to review the ICS-CERT destructive malware white paper for detailed mitigation guidance. It is important for organizations to maintain backups of key data, systems, and configurations such as:

- Server gold images,
- ICS Workstation gold configurations,
- Engineering workstation images,
- PLC/RTU configurations,
- Passwords and configuration information, and
- Offline copies of install media for operating systems and control applications.

Ensure Proper Configuration/Patch Management

Adversaries often target unpatched systems. A configuration/patch management program centered on the safe importation and implementation of trusted patches will help render control systems more secure.

Such a program will start with an accurate baseline and asset inventory to track what patches are needed. The program will prioritize patching and configuration management of "PC-architecture" machines used in HMI, database server, and engineering workstation roles, as current adversaries have significant cyber capabilities against these systems. Infected laptops are a significant malware vector. Such a program will limit the connection of external laptops to the control network and ideally supply vendors with known-good company laptops. The program will also encourage initial installation of any updates onto a test system that includes malware detection features before the updates are installed on operational systems.

NCCIC recommends that operators:

- Use best practices when downloading software and patches destined for their control network;
- Take measures to avoid watering hole attacks;
- Use a web Domain Name System (DNS) reputation system;
- Obtain and apply updates from authenticated vendor sites;
- Validate the authenticity of downloads;
- Insist that vendors digitally sign updates, and/or publish hashes via an out-of-bound communications path, and only use this path to authenticate;
- Never load updates from unverified sources; and
- Reduce your attack surface area.

To the greatest extent possible, NCCIC recommends that operators:

- Isolate ICS networks from any untrusted networks, especially the Internet;
- Lock down all unused ports;
- Turn off all unused services; and
- Only allow real-time connectivity to external networks if there is a defined business requirement or control function.
    - If one-way communication can accomplish a task, operators should use optical separation (“data diode”).
    - If bidirectional communication is necessary, operators should use a single open port over a restricted network path.

Build a Defendable Environment

Building a defendable environment will help limit the impact from network perimeter breaches. NCCIC recommends operators segment networks into logical enclaves and restrict host-to-host communications paths. This can prevent adversaries from expanding their access, while allowing the normal system communications to continue operating. Enclaving limits possible damage, as threat actors cannot use compromised systems to reach and contaminate systems in other enclaves. Containment provided by enclaving also makes incident cleanup significantly less costly.

If one-way data transfer from a secure zone to a less secure zone is required, operators should consider using approved removable media instead of a network connection. If real-time data transfer is required, operators should consider using optical separation technologies. This allows replication of data without placing the control system at risk.

Additional details on effective strategies for building a defendable ICS network can be found in the ICS-CERT Defense-in-Depth Recommended Practice.

Implement Secure Remote Access

Some adversaries are effective at gaining remote access into control systems, finding obscure access vectors, even “hidden back doors” intentionally created by system operators. Operators should remove such accesses wherever possible, especially modems, as these are fundamentally insecure.
Operators should:

- Limit any accesses that remain;
- Where possible, implement “monitoring only” access enforced by data diodes, and not rely on “read only” access enforced by software configurations or permissions;
- Not allow remote persistent vendor connections into the control network;
- Require any remote access to be operator controlled, time limited, and procedurally similar to “lock out, tag out”;
- Use the same remote access paths for vendor and employee connections; do not allow double standards; and
- Use two-factor authentication if possible, avoiding schemes where both tokens are similar and can be easily stolen (e.g., password and soft certificate).

Monitor and Respond

Defending a network against modern threats requires actively monitoring for adversarial penetration and quickly executing a prepared response. Operators should:

- Consider establishing monitoring programs in the following key places: at the Internet boundary; at the business to Control DMZ boundary; at the Control DMZ to control LAN boundary; and inside the Control LAN;
- Watch IP traffic on ICS boundaries for abnormal or suspicious communications;
- Monitor IP traffic within the control network for malicious connections or content;
- Use host-based products to detect malicious software and attack attempts;
    - Use login analysis (e.g., time and place) to detect stolen credential usage or improper access, verifying all anomalies with quick phone calls;
    - Watch account and user administration actions to detect access control manipulation;
- Have a response plan for when adversarial activity is detected; and
    - Such a plan may include disconnecting all Internet connections, running a properly scoped search for malware, disabling affected user accounts, isolating suspect systems, and immediately resetting 100 percent of passwords.
    - Such a plan may also define escalation triggers and actions, including incident response, investigation, and public affairs activities.
- Have a restoration plan, including “gold disks” ready to restore systems to known good states.

Reference

https://www.us-cert.gov/ncas/alerts/TA17-163A