Threat Research

TRITON Actor TTP Profile, Custom Attack Tools, Detections, and ATT&CK Mapping

April 10, 2019 | by Steve Miller, Nathan Brubaker, Daniel Kapellmann Zafra, Dan Caban

MALWARE TTPS ICS SECURITY

Overview

FireEye can now confirm that we have uncovered and are responding to **an additional intrusion by the attacker behind TRITON at a different critical infrastructure facility**.

In December 2017, FireEye publicly released our first analysis on the TRITON attack where malicious actors used the TRITON custom attack framework to manipulate industrial safety systems at a critical infrastructure facility and inadvertently caused a process shutdown. In subsequent research we examined how the attackers may have gained access to critical components needed to build the TRITON attack framework. In our most recent analysis, we attributed the intrusion activity that led to the deployment of TRITON to a Russian government-owned technical research institute in Moscow.

The TRITON intrusion is shrouded in mystery. There has been some public discussion surrounding the TRITON framework and its impact at the target site, yet little to no information has been shared on the tactics, techniques, and procedures (TTPs) related to the intrusion lifecycle, or how the attack made it deep enough to impact the industrial processes. The TRITON framework itself and the intrusion tools the actor used were built and deployed by humans, all of whom had observable human strategies, preferences, and conventions for the custom tooling of the intrusion operation. It is our goal to discuss these adversary methods and highlight exactly how the developer(s), operator(s) and others involved used custom tools in the intrusion.

In this report we continue our research of the actor's operations with a specific focus on a selection of custom information technology (IT) tools and tactics the threat actor leveraged during the early stages of the targeted attack lifecycle (Figure 1). The information in this report is derived from multiple TRITON-related incident responses carried out by FireEye Mandiant.

Using the methodologies described in this post, FireEye Mandiant incident responders have uncovered additional intrusion activity from this threat actor – including new custom tool sets – at a second critical infrastructure facility. As such, we strongly encourage industrial control system (ICS) asset owners to leverage the indicators, TTPs, and detections included in this post to improve their defenses and hunt for related activity in their networks.

For IT and operational technology (OT) incident response support, please contact FireEye Mandiant. For more in-depth analysis of TRITON and other cyber threats, consider subscribing to FireEye Cyber Threat Intelligence.

FireEye's SmartVision technology, which searches for attackers during lateral movement activities by monitoring east-west traffic in IT and OT networks, reduces the risk of an attack reaching sensitive ICS processes. This is particularly relevant for sophisticated ICS-related intrusions as attackers typically move from corporate IT to OT networks through systems that are accessible to both environments, far beyond perimeter defenses.

Contents

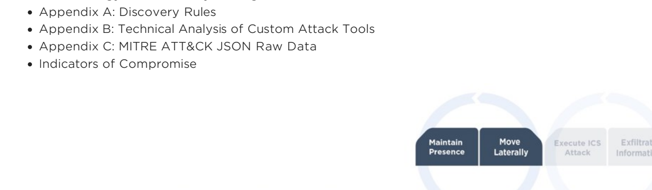
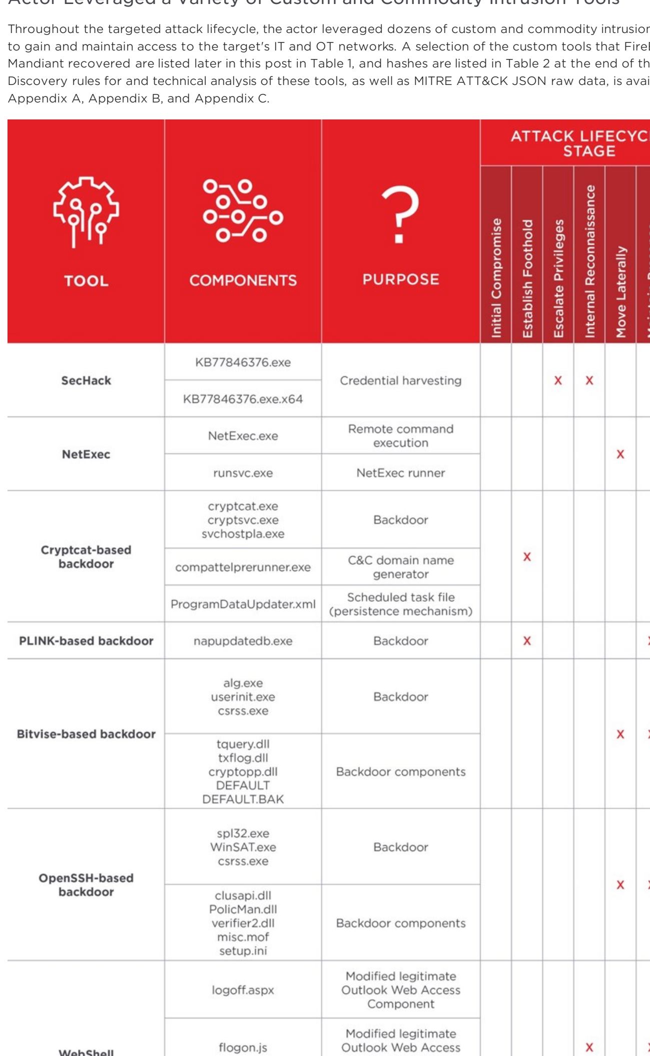


Figure 1: The FireEye targeted attack lifecycle

Actor Leveraged a Variety of Custom and Commodity Intrusion Tools

Throughout the targeted attack lifecycle, the actor leveraged dozens of custom and commodity intrusion tools to gain and maintain access to the target's IT and OT networks. A selection of the custom tools that FireEye Mandiant recovered are listed later in this post in Table 1, and hashes are listed in Table 2 at the end of this post. Discovery rules for and technical analysis of these tools, as well as MITRE ATT&CK JSON raw data, is available in Appendix A, Appendix B, and Appendix C.



TOOL	COMPONENTS	PURPOSE	Initial Compromise	Establish Foothold	Escalate Privileges	Internal Reconnaissance	Move Laterally	Maintain Presence	Complete Mission
SecHack	KB77846376.exe	Credential harvesting			X	X			
	KB77846376.exe.x64								
NetExec	NetExec.exe	Remote command execution					X		
	runsvc.exe	NetExec runner							
Cryptcat-based backdoor	cryptcat.exe cryptsvc.exe svchostpla.exe	Backdoor		X					
	compattelprerunner.exe	C&C domain name generator							
	ProgramDataUpdater.xml	Scheduled task file (persistence mechanism)							
PLINK-based backdoor	napupdatedb.exe	Backdoor		X				X	
Bitvise-based backdoor	alg.exe userinit.exe csrss.exe	Backdoor					X	X	
	tquery.dll txflog.dll cryptopp.dll DEFAULT DEFAULT.BAK	Backdoor components							
OpenSSH-based backdoor	spl32.exe WinSAT.exe csrss.exe	Backdoor					X	X	
	clusapi.dll PolicMan.dll verifier2.dll misc.mof setup.ini	Backdoor components							
WebShell	logoff.aspx	Modified legitimate Outlook Web Access Component				X		X	
	flogon.js	Modified legitimate Outlook Web Access Component							
	ftpexts.tlb	Output file containing credentials harvested by logoff.aspx							

Figure 2: Selection of custom tools used by the actor

The actor's custom tools frequently mirrored the functionality of commodity tools and appear to be developed with a focus on anti-virus evasion. The group often leveraged custom tools when they appeared to be struggling with anti-virus detection or were at a critical phase in the intrusion (e.g., they switched to custom backdoors in IT and OT DMZ right before gaining access to the engineering workstation). In some instances, the actor leveraged custom and commodity tools for the same function. For example, they used Mimikatz (public) and SecHack (custom) for credential harvesting; both tools provide a very similar output (Figure 2).

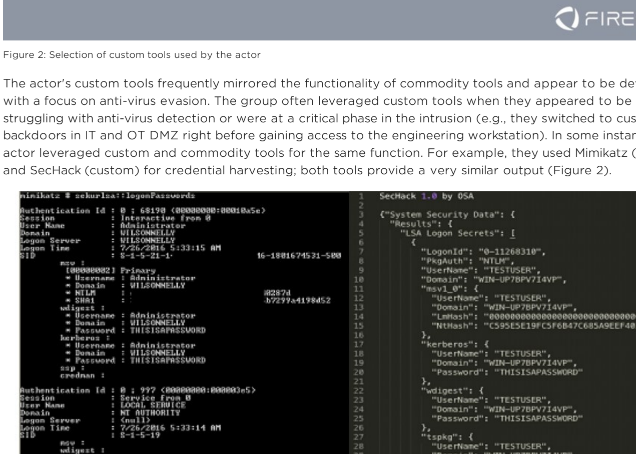


Figure 3: Default outputs for Mimikatz (left) and SecHack (right)

Tools and TTPs Indicate a Deep Interest in Ensuring Prolonged and Persistent Access to the Target Environment

The targeted attack lifecycle of a sophisticated ICS attack is often measured in years. Attackers require a long time to prepare for such an attack in order to learn about the target's industrial processes and build custom tools. These attacks are also often carried out by nation states that may be interested in preparing for contingency operations rather than conducting an immediate attack (e.g., installing malware like TRITON and waiting for the right time to use it). During this time, the attacker must ensure continued access to the target environment or risk losing years of effort and potentially expensive custom ICS malware. This attack was no exception. The actor was present in the target networks for almost a year before gaining access to the Safety Instrumented System (SIS) engineering workstation. Throughout that period, they appeared to prioritize operational security.

After establishing an initial foothold on the corporate network, the TRITON actor focused most of their effort on gaining access to the OT network. They did not exhibit activities commonly associated with espionage, such as using key loggers and screenshot grabbers, browsing files, and/or exfiltrating large amounts of information. Most of the attack tools they used were focused on network reconnaissance, lateral movement, and maintaining presence in the target environment.

The actor used multiple techniques to hide their activities, cover their tracks, and deter forensic examination of their tools and activities.

- They renamed their files to make them look like legitimate files, for example, KB77846376.exe, named after Microsoft update files.
- They routinely used standard tools that would mimic legitimate administrator activities. This included heavy use of RDP and PsExec/WinRM.
- When planting webshells on the Outlook Exchange servers, they modified already existing legitimate flogon.js and logoff.aspx files.
- They relied on encrypted SSH-based tunnels to transfer tools and for remote command/program execution.
- They used multiple staging folders and opted to use directories that were used infrequently by legitimate users or processes.
- They routinely deleted dropped attack tools, execution logs, files staged for exfiltration, and other files after they were finished with them.
- They renamed their tools' filenames in the staging folder so that it would not be possible to identify the malware's purpose, even after it was deleted from the disk through the residual artifacts (e.g., ShimCache entries or WMI Recently Used Apps).
- They used timestomping to modify the $STANDARD_INFORMATION attribute of the attack tools.

Once the actor gained access to the targeted SIS controllers, they appeared to focus solely on maintaining access while attempting to successfully deploy TRITON. This involved strategically limiting their activities to mitigate the risk of being discovered.

- The actor gained a foothold on the distributed control system (DCS) but did not leverage that access to learn about plant operations, exfiltrate sensitive information, tamper with the DCS controllers, or manipulate the process.
- They then gained access to an SIS engineering workstation. From this point forward, they focused most of their effort on delivering and refining a backdoor payload using the TRITON attack framework.
- They attempted to reduce the chance of being observed during higher-risk activities by interacting with target controllers during off-hour times. This would ensure fewer workers were on site to react to potential alarms caused by controller manipulation.
- They renamed their files to make them look like legitimate files, for example, trilog.exe, named after a legitimate Schneider Electric application.

Operational Since At Least 2014

Based on analysis of the actor's custom intrusion tools, the group has been operating since as early as 2014. It is worth noting that FireEye had never before encountered any of the actor's custom tools, despite the fact that many of them date to several years before the initial compromise. This fact and the actor's demonstrated interest in operational security suggests there may be other target environments – beyond the second intrusion announced in this blog post – where the actor was or still is present.

- A sample of a Cryptcat-based backdoor used to establish the initial foothold was recovered during the investigation; the sample was compiled and uploaded to a malware testing environment by the actor in 2014.
- Cryptcat- and PLINK-based backdoors were scheduled to execute daily starting from April 28, 2014, using ProgramDataUpdater and NetworkAccessProtectionUpdateDB tasks. This date is unrelated to the observed intrusion timeline and may indicate the date the threat actors first created these persistence mechanisms.
- NetExec.exe, a custom lateral movement and remote command execution tool, is self-titled "NetExec 2014 by OSA."
- SecHack.exe "by OSA," a custom credential harvesting and reconnaissance tool, was compiled on Oct. 23, 2014.
- The attackers used a pirated version of Wii.exe, a public file indexing tool that came with a license from 2010 and has not been updated since 2014.

ICS Asset Owners Should Prioritize Detection and Defense Across Windows Systems in Both IT and OT

Most sophisticated ICS attacks leveraged Windows, Linux, and other traditionally "IT" systems (located in either IT or OT networks) as a conduit to the ultimate target. Some examples include leveraging computers to gain access to targeted PLCs (e.g., Stuxnet), interacting directly with internet-connected human machine interfaces (HMIs) (e.g., BlackEnergy), and gaining remote access to an engineering station to manipulate a remote terminal unit (RTU) (e.g., INDUSTROYER) or infect SIS programmable logic controllers (PLC) (e.g., TRITON).

Defenders who focus on stopping an attacker in these "conduit" systems benefit from a number of key advantages. These advantages will only grow as IT and OT systems continue to converge.

- Attackers commonly leave a broad footprint in IT systems across most if not all the attack lifecycle.
- It is ideal to stop an attacker as early in the attack lifecycle as possible (aka "left of boom"). Once an attacker reaches the targeted ICS, the potential of a negative outcome and its severity for the target increase dramatically.
- There are many mature security tools, services, and other capabilities already available that can be leveraged to defend and hunt in "conduit" systems.

Leveraging Known Tools and TTPs To Hunt For the TRITON Actor

Historic activity associated with this actor demonstrates a strong development capability for custom tooling. The developer(s) behind these toolsets leaned heavily on existing software frameworks and modified them to best serve the intrusion operations. The developer(s) had preferences regarding the ports, protocols, persistence mechanisms, and other aspects of how the malware operated.

While the preferences of the development team supporting this activity will likely shift and change over time, learning about them is still useful to identify whether their TTPs are applicable to other malware developers and threat actors. Additionally, the actor possibly gained a foothold on other target networks—beyond the two intrusions discussed in this post – using similar strategies. In such cases, retrospective hunting would help defenders identify and remediate malicious activity.

Based on the examination of developer(s) preferences and abstracted adversary methodologies, it is possible to build broader visibility of the TTPs using detection and hunting rules of various fidelity and threat density. The compilation of these rules makes it possible to identify and classify potentially malicious samples while building new "haystacks" in which to hunt for adversary activity.

The TTPs we extracted from this actor's activities are not necessarily exclusive, nor are they necessarily malicious in every circumstance. However, the TTP profile built by FireEye can be used to search for patterns of evil in subsets of network and endpoint activity. Not only can these TTPs be used to find evidence of intrusions, but identification of activity that has strong overlaps with the actor's favored techniques can lead to stronger assessments of actor association, further bolstering incident response efforts.

The following table provides insights into notable methodologies surrounding the use of custom tools and tips for identifying evidence of this and related activity. Adversary methodologies are also expressed in terms of the MITRE ATT&CK framework (see Appendix C for MITRE ATT&CK JSON raw data).

Adversary Methodology	Discovery Tips
Persistence by Scheduled Tasks by XML trigger ATT&CK: T1053	Look for new and anomalous Scheduled Tasks XML triggers referencing unsigned .exe files.
Persistence by IFEO injection ATT&CK: T1183	Look for modifications and new entries referencing .exe files under registry key HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options.
Command and control (C2) established using hard-coded DNS servers	Look for PEs executions with run DNS lookups to 8.8.8.8:53. This may be applicable to sandbox and other malware processing technologies.
C2 using favored C2ports ATT&CK: T1043 ATT&CK: T1065	Look for outbound connections with port-protocol mismatches on common and uncommon ports such as 443, 4444, 8531, and 50501.
C2 using favored Virtual Private Server (VPS) infrastructure ATT&CK: T1329	Look for inbound and outbound connections from and to non-standard IP ranges, especially from international VPS providers like OVH and UK-2 Limited (uk2.net).
C2 domains with hyphen	Look for newly observed 2LD and 3LD domains that contain hyphens.
C&C using dynamic DNS domains from afraid.org ATT&CK: T1311	Look for newly observed dynamic DNS domains owned or registered with afraid.org.
C2 domains registered with vfemail.net email addresses	Look for newly observed domains or DNS resolutions to domains with registrant email information containing vfemail.net
Tunneled RDP using PLINK ATT&CK: T1076	Look for the presence of PLINK and non-standard RDP usage with event logs, firewall logs, and registry keys as described in the FireEye blog post "Bypassing Network Restrictions Through RDP Tunneling." Find internal RDP pivoting by looking for bitmap cache files under user accounts that should not be accessing sensitive systems via RDP. Look for bitmap cache files such as bcache22.bmc under default, service, or administrator accounts or any account not expected to be conducting internal RDP accesses to sensitive systems in a protected OT-connected zone, especially in the DMZ or DCS areas like HMIs or engineering workstations.
C2 using hard-coded SSH private keys	Look for PEs with hard-coded OpenSSH private keys.
Use of direct RDP ATT&CK: T1076	Look for inbound RDP connections with default host information, non-standard or unexpected locale IDs, or other metadata. See also the FireEye blog post on baselining RDP activity.
C2 using source systems with default Windows hostnames	Look for default Windows hostnames that fit the structure WIN-[A-Z0-9]{11} (e.g., WIN-ABCDEFGH1JK) in PE certificates, SSL and SSH certificates, and RDP handshakes.
C2 using SSH	Look for new, unique, or unusual SSH sessions. Logging of SSH keys and fingerprints would quickly and easily identify an anomalous session as a result of malware. Look for SSH over non-standard ports.
Compromised VPN accounts ATT&CK: T1078	Look for VPN logon anomalies based on infeasible patterns such as source account location, IP address, and hostname associations. Check out the FireEye blog post and free toolset for VPN logon analysis, GeoLogonalyzer. If you use SMS-based MFA, look for phone numbers registered outside the country where your employees operate.
Malware masquerading as Microsoft Corporation	Look for PEs with mismatched PE metadata such as contains "Bitvise" strings and also "Microsoft Corporation" in the metadata. Look for unsigned "Microsoft Corporation" binaries in the group's common staging directories.
Use of customized Bitvise binaries	Look for PEs with Bitvise PDB path strings such as d:\repos\main\ssh2\.
Use of customized OpenSSH binaries	Look for PEs with content "Microsoft openSSH client."
Use of customized Cryptcat but with default password	Look for PEs that drop Cryptcat binaries or contain Cryptcat string content such as the default password "metallica."
Timestomping via PowerShell ATT&CK: T1099	Look for timestomping command strings such as ".CreationTime=" in PowerShell scripts or in PowerShell command-line entries. Look for PEs with NTFS creation time prior to PE compile time.
Deployment of binaries with debug information from developer workstations with Visual Studio 2010	Look for PEs with PDB paths containing default or generic paths such as • \Users\user\Documents\Visual Studio 2010\ • \Documents\Visual Studio 2010\.
Use of Thinstall for packaging malware	Look for PE with content "thinstall\modules\boot_loader.pdb." Look for Thinstall binaries that have created virtualized files in the context of the SYSTEM user "C:\Windows\SysWOW64\config\systemprofile\AppData\Roaming\Thinstall\."
Use of favored directories for operating, staging and executing files	Look for new, unexpected, or otherwise anomalous binaries in the following directories: • C:\Windows\system32\inetsrv\ • C:\Windows\temp\ • C:\Windows\SysWOW64\wbem • C:\Windows\SysWOW64\drivers • C:\Windows\SysWOW64 • C:\Windows\system32\wbem\ • C:\Windows\system32\drivers\ • C:\Windows\system32\ • C:\Windows\ • C:\Users\Public\Libraries\ • C:\Users\administrator\appdata\local\temp\ • C:\ssh\ • C:\perflogs\admin\servermanager\ssh\ • C:\perflogs\admin\servermanager\ • C:\perflogs\admin\ • C:\perflogs\ • C:\cpqsystem\ • C:\hp\hpdiags\ • C:\hp\bin\log\

Table 1: TRITON actor methodology and discovery strategies

Outlook

There is often a singular focus from the security community on ICS malware largely due to its novel nature and the fact that there are very few examples found in the wild. While this attention is useful for a variety of reasons, we argue that defenders and incident responders should focus more attention on so-called "conduit" systems when trying to identify or stop ICS-focused intrusions.

In an attempt to raise community awareness surrounding this actor's capabilities and activities between 2014 and 2017—an effort compounded in importance by our discovery of the threat actor in a second critical infrastructure facility—we have shared a sampling of what we know about the group's TTPs and custom tooling. We encourage ICS asset owners to leverage the detection rules and other information included in this report to hunt for related activity as we believe there is a good chance the threat actor was or is present in other target networks.

For IT and OT incident response support, please contact FireEye Mandiant. For more in-depth analysis of TRITON and other cyber threats, consider subscribing to FireEye Cyber Threat Intelligence.

FireEye's SmartVision technology, which searches for attackers during lateral movement activities by monitoring east-west traffic in IT and OT networks, reduces the risk of an attack reaching sensitive ICS processes. This is particularly relevant for sophisticated ICS-related intrusions as attackers typically move from corporate IT to OT networks through systems that were accessible to both environments, far beyond perimeter defenses.

Appendices

- Appendix A: Discovery Rules
- Appendix B: Technical Analysis of Custom Attack Tools
- Appendix C: MITRE ATT&CK JSON Raw Data

Indicators of Compromise

Filename	Hash
KB77846376.exe	MD5: 47f9cc543905a69a423f9110ae7deffb SHA256: 87648aad45d9142d1d825d728b7aa098f92aea38698209d038ba58b7385f8df6
KB77846376.exe.x64	MD5: ee477fdee8b6ad4fe778a6fa4058f9aa SHA256: 2141b526a81bb87b964880e69933aad3932131ccccee5949d2a16c1e124ccdbb
Netexec.exe	MD5: aca94bb7bdfb735f267f083e28f4db37 SHA256: c55e63f8a3b328c3ba77cebf821bdc5243b15a0298057e75f7605d0922c8d7cd
runsvc.exe	MD5: 1904cad4927541e47d453becbd934bf0 SHA256: 70efbd074326e7bbd4e851ded5c362fe5fe06282ed4bbb4b9f761f1b12ee32f7
svchostpla.exe	MD5: 121772100e46dde2d6317b08c7a59e13 SHA256: 910b26c942c0cff8b1f5a57e1521801bfd54c8cbcfd23d3d11ea9fe27ca4a0e9
compattelprerunner.exe	MD5: 35f443608fc4eeb78f9347a9dfc5aea1 SHA256: 1330594c2685fe6fc2c87439ef151dfacabc78402379a73be39953048b144960
compattelprerunner.exe	MD5: 10fd713eb3bc6a8f7abd7030104d0ce7 SHA256: 6ab948ec61f1f7e04119da85d5263d428a1de070edad3a4e796bada2ab05cea7
compattelprerunner.exe	MD5: 648223034bda28c415a8deeb74dcb3ef SHA256: 4c2383c8650112e00cb8b52d0faac7b98207073db081dbdcbb278f0470b869a1
ProgramDataUpdater.xml	MD5: c744006ebaaf25cd7fad0ebba56e4f84 SHA256: 6d2d9623762f822949eef80b02f4ba2d26227eb23ad5b8d1a0a3d6da3bc60d6c
napupdatedb.exe	MD5: ba51f25db03a66c658d1fd4396f32843 SHA256: 0fc391cdef0705f032109e16f8f591e1e6f8ffccbc46f4eb4a8fa058047c0adc
alg.exe	MD5: af5b9c9e4c6bfc6cb7fa5e4b04da8dc8 SHA256: 970fab66733ba594b435cf345c72814ee5f8443c44d28ef251f768ad66a6c052
userinit.exe	MD5: 2d11be6755b80cfca5c2f5138881ff25 SHA256: fc5b4c61f66beb58a62636ab7c198e6ab7f38ce201f098f2818a5699b8aa1138
csrss.exe	MD5: de2e1d59c81a2798a239baaa1edc0dd8 SHA256: 1848d26e47ee4937ef02e67a447b4054d66f4d659f1fbd8bda1482dc4f02c7c4
tquery.dll	MD5: 31cd0738ec2e40ff086dfd84ac2510fb SHA256: 98da0ce88de897e1b08733ac771edab5e5b2a2dda8aab0e73c1d41bade275ff6
txflog.dll	MD5: 8db693f75a0cfe043a5810f799654cf9 SHA256: f0dcbc83d911c382da7ba06a027bdd5861d1a9b723ebe5d9d6f6b79d7b70f29d
cryptopp.dll	MD5: 9a7234078559093e06c9d32148ed95a3 SHA256: 32f5d0a454c26e8aa6f4cad58f3782337cc97cfe2305bbfe564437e5f0d51bbc
DEFAULT	MD5: 30a9ee20052fcc34dee6b09f9210d4ed SHA256: f7bdddbeae239305ccca3b7eb1019b713bd0f7f060976494e810917a1e6ad5ee
DEFAULT.BAK	MD5: 519098f3970d57b8429a9f6baeaf0f8c SHA256: 1f1902e4482527824ef2c0c2039162db85e5a671caf0767a695116b03cfc866d
spl32.exe	MD5: 62831f960fe764f090d1201033202438 SHA256: 1d359163b6bd882ae4c26854d69745136a23f3abb7c96341f6d17e18a546a5fd
WinSAT.exe	MD5: 685776e0020ad9bfc4e2f4f7c7a9c623 SHA256: 3b6fd091b956b17476990c6ca77dd8f77d203d3170745d1b7c7894bfcf629b86
csrss.exe	MD5: d05702c4c3924b08bac5079add4e2347 SHA256: 720ef3d5b5416974376ca4ea8bd536e9eeb608f89e3b5b264e197266be8a9f4e
clusapi.dll	MD5: f985fd0d36ab79bfccfaed6d64c5fc23 SHA256: 084c21e75fbfa5056fec913c237ce7fba314f88fbd687e8dcb1e777003f79b0e
PolicMan.dll	MD5: 6fbeb6a9f990402bf6f056c892fefcc6 SHA256: 9224c2b00e94e5c57d63820aebe613843b5c851a027488148308fac2d02206f0
verifier2.dll	MD5: 6f8b33cb1d101c6bf0e9aeaf29b7e72d SHA256: 7633b4178611e28aedfa365a0de8ebe5f41ae8eeee71322f04d0e30e50ba2914
misc.mof	MD5: 5efbd51044fb90c6231438c51d83037f SHA256: 7bcca38e43f3b37b1acea05899a7c11dfb62de64531bd48af992d5e400a1755f
logoff.aspx	MD5: 915efc70a812c1cb35b29ba0ecb7c48d SHA256: 0da4c0b83fa1ad4af9aad6c42feecc6c21c3fd0e660b9e5b3857ddeae3473d54
flogon.js	MD5: 0f144e79ea8d8b66fa973e0568415501 SHA256: f81aa77d23ca6662efb3e6e33538a60e39abb5ca66102e07ffa318a6d6cd78ec

Table 2: Indicators of compromise

< PREVIOUS POST
NEXT POST >