Cyberespionage campaign hits energy companies

SilentDefense helps detecting and mitigating the threat

Authors

Joel Langill - Industrial Cyber Security Expert, Founder of SCADAhacker.com
Emmanuele Zambon, PhD - SecurityMatters Founder and CTO
Daniel Trivellato, PhD - SecurityMatters Product Manager

8 July 2014

Contents

Preface

Over the past couple of weeks, cybersecurity vendors [1, 2, 4] have announced the uncovering of a successful cyberespionage campaign carried out by the Dragonfly hacking group. In the most recent string of attacks, Dragonfly has targeted multiple US and European energy companies, successfully looting valuable process information in what appears to be the next step in the cyberwarfare campaign against critical infrastructure organizations, after Stuxnet in 2010. Cybersecurity vendors have scrutinized the campaign and presented an analysis of the malware employed by Dragonfly to steal information from the infected computers. This short paper revisits the main points of this investigation and illustrates why the implementation of a defense-in-depth strategy is key to successfully counter cyberthreats like Dragonfly.

The Dragonfly cyberespionage campaign

The Dragonfly hacker group has successfully mounted a cyberespionage operation against US and European companies, mainly in the energy sector. The group managed to install a remote access tool (RAT) in computers used for running Industrial Control Systems (ICS), and to harvest data from the infected machines utilizing a payload designed for a specific industrial protocol. According to Symantec [1], the Dragonfly campaign appears to have a much broader focus than the preceding Stuxnet campaign: "While Stuxnet was narrowly targeted at the Iranian nuclear program and had sabotage as its primary goal, Dragonfly appears to have a much broader focus with espionage and persistent access as its current objective with sabotage as an optional capability if required." There has been no proof that any sabotage capabilities were used by the Dragonfly group to date, but capabilities may exist in the toolkits employed, representing possibly the scariest part of the story, as they could potentially open doors to dramatic scenarios. Was the stealing of industrial information from energy companies only the first step of a destructive cyberwarfare campaign?

The attacker

The Dragonfly hacker group (also known as Energetic Bear) appears to be in operation since 2011. It initially targeted organizations in the defense and aviation industries in US and Canada, before moving their attention to US and Western European energy firms in 2013. An analysis of the malware code used in the campaign has shown that the group worked mostly during Eastern European working hours (Monday through Friday from 9AM to 6PM, UTC+4 time zone), suggesting that most group members worked in that region. The complexity of the operation leads many to believe that Dragonfly is a well-funded, possibly state-sponsored group of adversaries.

The targets

So far, the campaign has resulted in the leakage of information from multiple organizations, many of which operate in the energy sector, and range from electricity generation companies, electricity grid and petroleum pipeline operators, and industrial system and equipment providers. The majority of the victims are located in Europe, followed by the US. Figure 1 shows the top 10 countries by active infections (i.e. where the attacker has extracted information from infected computers).

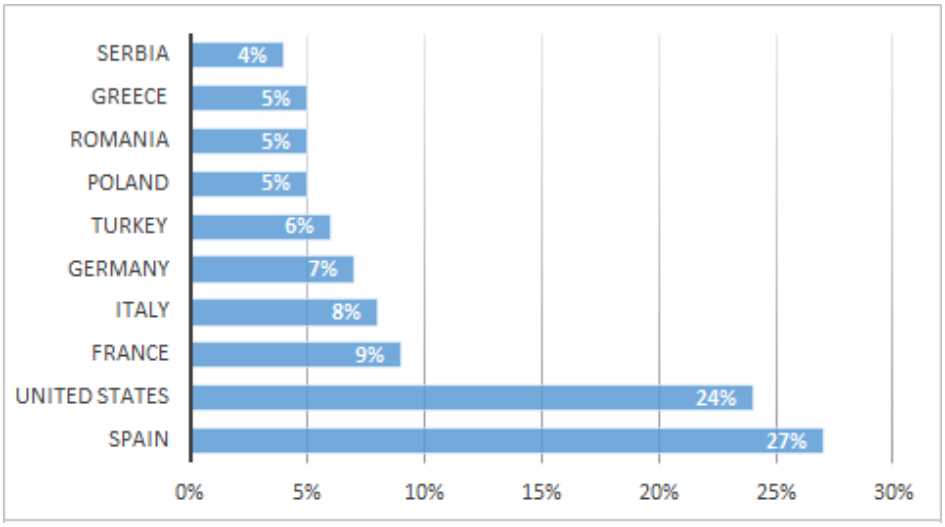

Figure 1: Top 10 countries by active infections [1]

In total, the number of infected machines that attempted to report to a malware command and control (C&C) server is approximately 1500 [2]. These numbers represent hosts that have been compromised and have established C&C communications, which does not necessarily represent the number of actual industrial control system hosts compromised (this will be discussed in more detail later in this paper). Precise information about the extent of the ICS compromise is not available at the time of writing, but is expected to be significantly less than shown in the reported chart.

Attack vectors

Dragonfly used two pieces of malware in the attack; both were remote access tools (RATs) designed to carry out the cyberespionage operations. The RATs were distributed and reached the victims' machines through three attack vectors:

- **E-mail campaign**: selected executives and senior employees of targeted companies would receive e-mails with a malicious PDF attachment containing the RAT. Symantec identified seven different organizations targeted in this campaign; the number of emails sent to each organization ranged from 1 to 84 [1].
- **Watering hole attacks**: these attacks targeted a number of legitimate websites likely to be visited by individuals working in the energy sector. Upon visiting one of the infected websites, the visitor would be redirected to another compromised legitimate website hosting an exploit kit. This exploit kit would in turn install the RAT on the visitor's computer.
- **Software downloaded from ICS related vendors**: Dragonfly members managed to hack the websites of at least three different ICS related vendors, and insert malware

into the legitimate software they were making available for download to their clients. The malware would then be installed on the victim's computer upon download of the trusted software or update. The targeted vendors are based in Germany, Switzerland, and Belgium. The first identified software package to be Trojanized was used to provide VPN access to programmable logic controller (PLC) and similar devices. The second company manufactures a PLC type device, and had one of their communication drivers Trojanized. The third company included in this campaign develops ICS systems for primarily renewable energy markets.

Dragonfly employed these attack vectors in three successive phases of the campaign. The e-mail attacks were conducted between February and June 2013; they were followed by the watering hole attacks beginning in June 2013 that included the compromise of ICS vendor websites. The first ICS vendor website was compromised for a period of six weeks during the period of June-July 2013, followed by the second vendor in January 2014. This second vendor was able to identify the breach, notify affected users and mitigate the situation in about 10 days. It is estimated that around 250 downloads of the infected software occurred during this phase [3]. The final ICS vendor website breach occurred in April 2014 and lasted two weeks.

Malware operation

The attackers employed two RATs to steal information from the infected computers and send it to C&C servers under the control of the attackers. Both of these RATs provided the capability of downloading and executing files remotely via the C&C servers:

- **The Havex RAT** (also known as `Backdoor.Oldrea`): it allows the attacker to extract data from the Outlook address book and ICS related software configuration files used for remote access from the infected computer to other industrial systems. Furthermore, it gathers system information on the installed programs, local file lists, and available drives.
- **Karagany** (`Trojan.Karagany`): it allows the attackers to upload and download files from the infected computer and to run executable files. It also contains advanced features for collecting passwords, taking screenshots, and cataloguing documents stored on the victim's machine. Karagany was already available on the underground market, although the Dragonfly group might have modified the source code to best fits its purposes.

Most of the victims were infected with the Havex RAT; Karagany was identified on only 5% of the infected computers. Havex appears to be custom malware either written by the Dragonfly group itself or commissioned by them. Security analysts of F-Secure [2] have identified and analyzed 88 variants of Havex, which contacted 146 C&C servers to communicate the stolen information. The majority of the C&C servers host blogs and content management systems (primarily WordPress[1]), presumably compromised by the attackers using similar exploits. These numbers strengthen the belief that the operation is state-sponsored.

From an operational viewpoint, certain payloads deployed with the Havex RAT exhibit "ICS network sniffing" behavior. In particular, they attempt to enumerate and qualify the devices in the local area network, and send the results back to the C&C servers. An analysis of the malware executables highlights that the attackers were looking for OPC (Open Platform Communications[2]) servers. OPC is a real-time data exchange protocol that supports bidirectional reading/writing of process variables, but does not provide more advanced capabilities

[1] http://pastebin.com/raw.php?i=qCdMwtZ6

[2] OPC was renamed from "Object Linking and Embedding (OLE) for Process Control" to "Open Platform Communications" in November 2011.

like device configuration and firmware updates. OPC is a standard way for process control systems, applications and devices to interact with each other.

It is important to note that not all variants of the Havex RAT and its associated payloads contained the code used to enumerate OPC services on a network. The observed payloads containing the ICS (OPC) components are believed to have originated only via the Trojanized software downloads from the three mentioned ICS vendor websites. This conclusion is based on analysis of malware referenced by F-Secure [2] and obtained through VirusTotal[3]. This means that the actual number of compromised ICS systems is likely much less than the identified number of hosts/sites infected by the Havex malware along.

Figures 2 and 3 shows the relevant extract of the Havex payload containing the network and OPC enumeration code. Looking at the malware code, we can indeed see that it uses Microsoft Component Object Model (COM) interfaces to detect whether the machines identified during the network scan run OPC services. The two COM interfaces found in the code are the following:

- `IOPCServerList` (CLSID = 13486D51-4821-11D2-A494-3CB306C10000)
- `IOPCServerList2` (IID = 9DD0B56C-AD9E-43EE-8305-487F3188BF7A)

```
.text:10001B45          mov     eax, [esp+98h+var_78]
.text:10001B49          add     esp, 0Ch
.text:10001B4C          push    edi             ; dwCoInit
.text:10001B4D          push    edi             ; pvReserved
.text:10001B4E          mov     [esp+94h+pServerInfo.pwszName], eax
.text:10001B52          mov     [esp+94h+pResults.pIID], offset unk_10030C78 {9dd0b56c-ad9e-43ee-8305-487f3188bf7a}
.text:10001B5A          mov     [esp+94h+pResults.pItf], edi                 Interface ID: IOPCServerList2
.text:10001B5E          mov     [esp+94h+pResults.hr], edi
.text:10001B62          call    ds:CoInitializeEx
.text:10001B68          lea     eax, [esp+8Ch+pResults]
.text:10001B6C          push    eax             ; pResults
.text:10001B6D          xor     ebx, ebx
.text:10001B6F          inc     ebx
.text:10001B70          push    ebx             ; dwCount
.text:10001B71          lea     eax, [esp+94h+pServerInfo]
.text:10001B75          push    eax             ; pServerInfo
.text:10001B76          push    17h             ; dwClsCtx
.text:10001B78          push    edi             ; punkOuter
.text:10001B79          push    offset Clsid    ; Clsid       {13486d51-4821-11d2-a494-3cb306c10000}
.text:10001B7E          mov     [esp+0A4h+var_4], edi         Class ID: OPCServerList
.text:10001B85          call    ds:CoCreateInstanceEx
```

Figure 2: Extract from Havex Executable (taken from samples obtained from VirusTotal)

Address	Length	Type	String
".." .rdata:10030CD0	00000050	unicode	Programm was started at %02i:%02i:%02i\n
".." .rdata:10030D20	00000006	unicode	a+
".." .rdata:10030D28	0000002C	unicode	%02i:%02i:%02i.%04i:
".." .rdata:10030D58	00000098	unicode	**\n
".." .rdata:10030DF0	0000003E	unicode	Start finging of LAN hosts...\n
".." .rdata:10030E30	0000004C	unicode	Finding was fault. Unexpective error\n
".." .rdata:10030E7C	00000038	unicode	Was found %i hosts in LAN:\n
".." .rdata:10030EB4	00000028	unicode	Hosts was't found.\n
".." .rdata:10030EDC	00000022	unicode	\t\t\t\t\t%02i) [%s]\n
".." .rdata:10030F00	00000042	unicode	Start finging of OPC Servers...\n
".." .rdata:10030F44	00000036	unicode	Was found %i OPC Servers.\n
".." .rdata:10030F80	00000054	unicode	\t\t%i) [%s\\%s]\n\t\t\tCLSID: %s\n
".." .rdata:10030FD8	0000006E	unicode	\t\t\tUserType: %s\n\t\t\tVerIndProgID: %s\n
".." .rdata:10031048	00000038	unicode	\t\t\tOPC version support: %s\n
".." .rdata:10031080	00000054	unicode	OPC Servers not found. Programm finished\n

Figure 3: Extract from Havex Strings (taken from samples obtained from VirusTotal)

The fact that Dragonfly is gathering information about OPC servers and VPN connections to PLCs might indicate that the final objective is to gain access to the PLCs themselves, which would enable the attackers to change, damage or disrupt the critical processes run by the targeted organizations.

[3] https://www.virustotal.com/

SilentDefense detects the malware used by Dragonfly

SecurityMatters' flagship product SilentDefense ICS is capable of detecting Havex in multiple stages of its operation, immediately alerting the security team of the threat and enabling the targeted victim to mitigate it before damage is done or sensitive information is disclosed. In particular, SilentDefense ICS detects Havex both when it attempts to connect to the C&C server to download or upload files and information, and when it scans the network to enumerate devices. In the next paragraphs we illustrate how SilentDefense ICS would detect and alert about the Havex behavior.

The network Behavioral Blueprint

SilentDefense ICS is a network monitoring and intrusion detection system that automatically models normal and acceptable network behavior and alerts whenever some network devices perform activities that diverge from their intended operation. SilentDefense ICS operates in two phases. First, it analyzes network communications and generates the network Behavioral Blueprint™. The Behavioral Blueprint defines communication patterns, protocols, message types, message fields, and field values that are normal for the monitored process. A review of the Behavioral Blueprint immediately reveals network and system misconfiguration (e.g., rogue devices), unintended communications, and unusual field values employed in the network, in case Havex already infected some devices. After this setup phase, SilentDefense ICS can be used for continuous monitoring to detect whenever network devices perform unintended activities. In the case of Havex, these unintended activities are represented by the network scan and the communication with the C&C servers.

Figures 4 and 5 show some examples of network Behavioral Blueprints as displayed by SilentDefense ICS. The examples represent the two "types" of Behavioral Blueprint that SilentDefense ICS can generate. In particular, Figure 4 shows the model of normal network communications automatically generated by SilentDefense ICS' LAN Communication Profile (LAN CP) engine. The LAN CP reports the observed network communications in terms of

communication patterns, protocols, and protocol message types normally used by the devices in the network. Note that this includes details of which device has communicated using which (D)COM interfaces. The controls implemented by the LAN CP make sure that whenever a network device connects to an unusual IP address (e.g., the C&C server), or invokes a COM interface that it has never used before or is not supposed to be used (e.g., the ones used by Havex), SilentDefense ICS raises an alert.

```
class DCOM 0 { { mt { 0, 2, 11, 12, 16 } }, { if { { 99fcfec4-5260-101b-bbcb-00aa0021347a, 2 } } } }
class DCOM 1 { { mt { 0, 2 } }, { if {  } } }
class DCOM 2 { { mt { 0, 2, 11, 12, 16 } }, { if { { 99fcfec4-5260-101b-bbcb-00aa0021347a, 1 } } } }
class SMB 0 { { smb1-mt {  } }, { smb2-mt {  } }, { mt {  } }, { if {  } } }
class MODBUSTCP 0 { { fc { 3, 22 } } }
class MODBUSTCP 1 { { fc { 3, 16 } } }
class RFB 0 { { * } }

alert { * } : { * } ->> { * } : { * } using { TCP } with { * } @mergeBy { l7ProtoClass } @count { 1 }
allow { $SCADA_SERVER } : { $DCOM_PORT } ->> { $PLC_GROUP_1 } : { $MODBUS_PORT } using { TCP } with { { class MODBUSTCP 1 } } @mergeBy
allow { $SCADA_SERVER } : { $DCOM_PORT } ->> { $HMI } : { $DCOM_PORT } using { TCP } with { { class DCOM 1 } } @mergeBy { l7ProtoClass
allow { $SCADA_SERVER } : { * } ->> { $HMI } : { 135 } using { TCP } with { { class DCOM 2 } } @mergeBy { l7ProtoClass } @count { 21 }
allow { $SCADA_SERVER } : { $DCOM_PORT } ->> { $PLC_GROUP_2 } : { $MODBUS_PORT } using { TCP } with { { class MODBUSTCP 0 } } @mergeBy
allow { $HMI } : { $DCOM_PORT } ->> { $SCADA_SERVER } : { $DCOM_PORT } using { TCP } with { { class DCOM 1 } } @mergeBy { l7ProtoClass
allow { $HMI } : { * } ->> { $SCADA_SERVER } : { 135 } using { TCP } with { { class DCOM 0 } } @mergeBy { l7ProtoClass } @count { 25 }
```

Figure 4: An example of LAN Communication Profiler Behavioral Blueprint

Figure 5 shows the model of normal protocol usage automatically generated by SilentDefense ICS' Deep Protocol Behavior Inspection (DPBI) engine. This model presents all fields (e.g. message types) and field values that are normally used for a certain protocol within the analysed network in the form of a protocol tree. The depicted tree was built for the DCOM protocol. On the right of the tree, we show how for each protocol field it is possible to observe in detail and edit the values observed. Again, the DPBI engine guarantees that if network devices use unusual (D)COM fields, the security team will immediately be alerted.

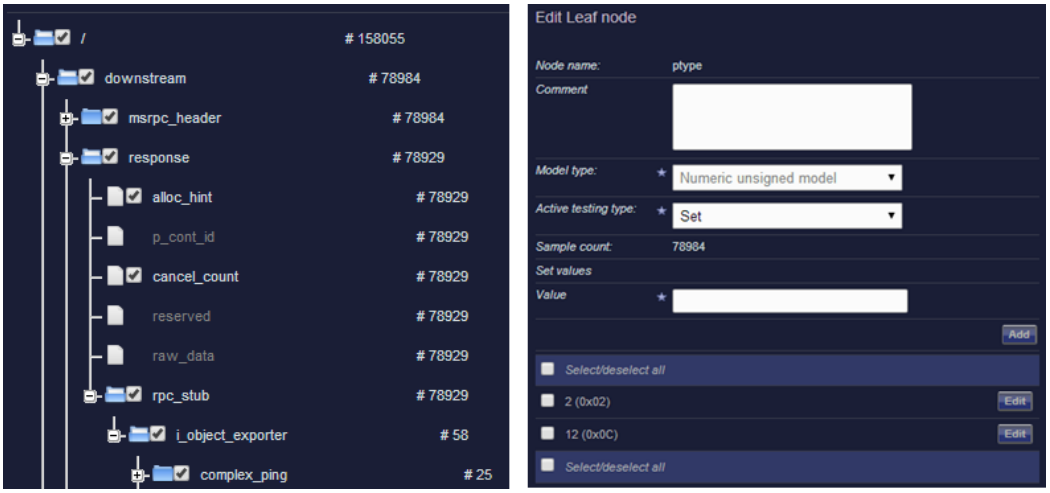

Figure 5: An example of Deep Protocol Behavioral Inspection Behavioral Blueprint

Detection of network scan

SilentDefense ICS can detect Havex through both LAN CP and DPBI engines. More precisely, the LAN CP might raise an alert for two types of unusual network activities. The first is the network scan performed by the infected machine (see Figure 3). In fact, when scanning the network, the infected machine might connect to devices with which it normally does not communicate, or is not supposed to communicate. The second unusual activity is the invocation of the `IOPCServer` COM interfaces. These interfaces are typically used only when ICS software is installed or updated on a certain device; at SecurityMatters' customers, we have never observed the use of these interfaces during normal operations. Their invocation

would thus result in alert being generated by SilentDefense ICS. If the invocation of these interfaces is followed by a communication with an unknown external IP address (e.g. the C&C server - as described in the following section), the host originating the communication is likely to be infected by Havex.

Figure 6 shows an example of an alert generated by the LAN CP when it detects the use of unusual COM interfaces. The unusual interfaces are indicated on the right: they are highlighted in red and marked with a warning sign. On the left, the alert reports details of the devices involved in the communication. This allows to immediately spot any devices infected by Havex (the "source" device).

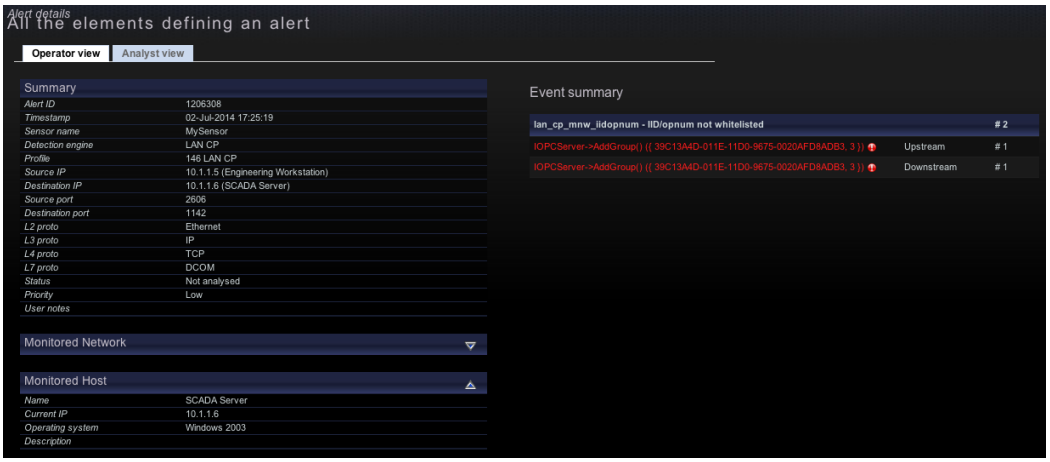

Figure 6: Alert generated by SilentDefense ICS when Havex invokes the COM interfaces to identify OPC servers

Similarly, the DPBI engine would detect and report the use of unusual COM protocol fields (and/or values) by any infected device. The alert generated by the DPBI engine would indicate the "source" of the unusual communication, thus identifying the infected machine, and the branch of the protocol tree that is not part of normal operations.

Detection of communication with C&C server

Further to detecting the network scan, SilentDefense ICS' LAN CP engine would alert the security team also whenever the infected machine attempts to communicate with the malware C&C server. In fact, the infected machine would connect to an IP address that is not "whitelisted" in the LAN CP model. This enables the security team to stop the communication before any sensitive network information is leaked, for instance by "blacklisting" the C&C IP in the company firewall.

Figure 7 illustrates an example alert generated by the LAN CP engine when a network device contacts an unusual IP address. The alert highlights in red and marks with a warning the unusual IP address, which can be immediately blacklisted if not recognized by the security team. Figure 7 illustrates an example alert generated by the LAN CP engine when a network device contacts an unusual IP address. The alert highlights in red and marks with a warning the unusual IP address, which can be immediately blacklisted if not recognized by the security team.

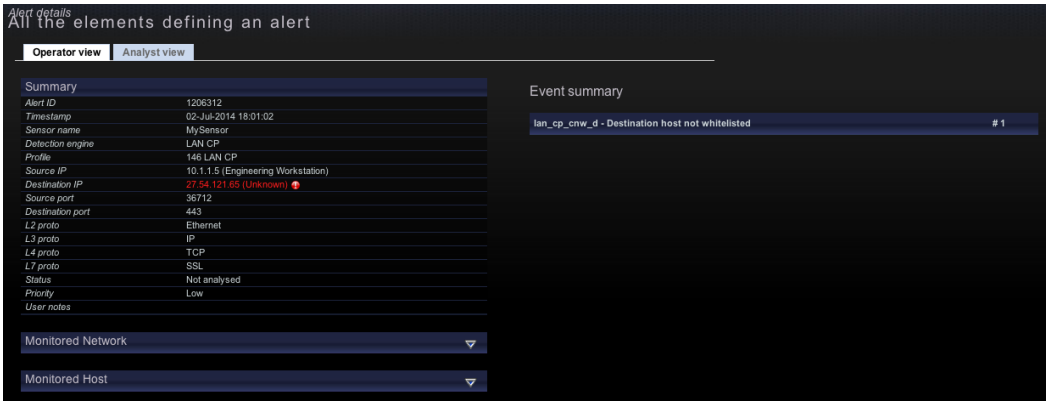

Figure 7: Alert generated by SilentDefense ICS when Havex communicates with the C&C server

Conclusions and recommendations

The Dragonfly hacker group is carrying out a cyberespionage campaign against energy companies in the US and Europe. So far, the campaign has resulted in the successful looting of strategic information from the energy companies' networks. The malware employed in the campaign, however, may give the attackers the capability to launch subsequent attacks with greater consequences.

It is fundamental that critical infrastructure organizations start adopting more progressive countermeasures to today's cyberthreats. The waiting time is over - it has been demonstrated more than once that skillful attackers can easily penetrate critical infrastructure networks, with the potential of causing immeasurable damages to the economy, security, and public safety and health of a country.

We believe that the implementation of a defense-in-depth strategy is key to successfully counter the increasing cyberthreat. The first defensive layer is of course represented by firewalls and/or intrusion prevention systems, which keep out of a network the *known* and easy-to-spot attacks. Cybersecurity vendors have already released signatures for their intrusion prevention and host-based solutions to detect and stop the malware used by Dragonfly. As indicated by F-Secure, however, 88 variants of the Havex malware have been identified so far. Signatures might not offer protection to new variants released by Dragonfly, or to the next malware employed in their campaign. It is therefore vital that along with traditional cybersecurity solutions enterprises deploy a non-signature based network monitoring solution like SilentDefense ICS, which does not rely on the knowledge of a threat to detect it and report it. SilentDefense ICS is unique in its kind, as no other solution is capable of automatically defining "normal network operations", and of analysing communications down to the values exchanged by network devices. This unique approach ensures protection from today's as well as tomorrow's threats.

Acknowledgments

We thank Damiano Bolzoni and Cliff Gregory for their useful insights.

About SecurityMatters

SecurityMatters is an international company with business in many critical infrastructure and industrial automation industries. Its research and development team works with many different projects throughout the EU and USA and delivers game-changing network monitoring and intrusion detection technology to make their customers more secure and in control.

About the Authors

Joel Langill Joel Langill is an Industrial Cyber Security Expert with over 30 years of field experience and is the founder of the globally recognized website SCADAhacker.com. He brings a unique perspective to industrial security having spent over three decades deploying ICS solutions covering most major industry sectors in over 35 countries encompassing all generations of automated control from pneumatic to cloud-based services. He has been directly involved in the specification and design of automation solutions spanning front-end engineering design, detailed design, system integration, commissioning, and legacy system migration. Joel currently provides a range of services to ICS end-users, system integrators, and governmental agencies worldwide. He works closely with suppliers in both consulting and R&D roles, and has developed a specialized training curriculum focused on applied ICS security. He served as co-author and technical editor for several books on industrial security. Joel serves on the Board of Directors for the Milwaukee Chapter of InfraGard, and is an ICS research focal point to numerous CERT organizations around the world. Joel was an active contributor to the research conducted relating to the impact of Heartbleed on ICS, was a key technical resource during the Stuxnet crisis, and has been credited with several coordinated disclosures relating to industrial automation and control.

Emmanuele Zambon Emmanuele Zambon has been involved in computer security since 2003. In 2005, he also had graduated in Computer Science from the Ca' Foscari University of Venice with a thesis on Intrusion Detection Systems. During and after his studies he has been employed in the Information Risk Management group of KPMG and in Telecom Italia as a consultant, doing ethical hacking, network vulnerability/assessment, and software development/performance tuning. In 2006, Zambon followed Bolzoni to the Netherlands to work together on the development of a new technology (the core of which now forms SilentDefense). Zambon received his PhD in IT Risk Management in 2011 from the University of Twente. He is now working part-time as a post-doc at the University of Twente, doing research on intrusion detection for industrial process automation networks, and working part-time for SecurityMatters.

Daniel Trivellato Daniel Trivellato pursued his Master's degree in Computer Science at the Free University of Bozen-Bolzano, Italy, graduating cum laude in 2007. In 2012, Daniel received his PhD in computer security from the Technische Universiteit Eindhoven, where he worked on the design and implementation of innovative access control solutions for dynamic, distributed, heterogeneous systems. His work was carried out under the supervision of prof. dr. Sandro Etalle. Since 2012, Daniel works as a project leader at SecurityMatters.

Bibliography

[1] Symantec Security Response Official Blog. Dragonfly: Western energy companies under sabotage threat. http://www.symantec.com/connect/blogs/dragonfly-western-energy-companies-under-sabotage-threat.

[2] F-Secure News from the Lab. Havex hunts for ics/scada systems. http://www.f-secure.com/weblog/archives/00002718.html.

[3] Talk2M. Incident report. http://www.talk2m.com/en/full_news.html?cmp_id=7&news_id=51.

[4] Ars Technica. Active malware operation let attackers sabotage us energy industry. http://arstechnica.com/security/2014/06/active-malware-operation-let-attackers-sabotage-us-energy-industry/.