Generating Application Layer IDS Rules from Cyber Threat Intelligence

A thesis submitted to the
Graduate School of Natural and Applied Sciences

by

Bahruz JABIYEV

in partial fulfillment for the
degree of Master of Science

in

CyberSecurity Engineering

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science in CyberSecurity Engineering.

APPROVED BY:

Necati Ersen Şişeci (Thesis Advisor)	.....
Prof. Dr. Tahsin Erkan Türe	.....
Doç. Dr. Oğuzhan Külekçi	.....

This is to confirm that this thesis complies with all the standards set by the Graduate School of Natural and Applied Sciences of İstanbul Şehir University:

DATE OF APPROVAL: 20 June 2016

SEAL/SIGNATURE:

Declaration of Authorship

I, Bahruz Jabiyev, declare that this thesis titled, 'Generating Application Layer IDS Rules from Cyber Threat Intelligence' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

Generating Application Layer IDS Rules from Cyber Threat Intelligence

Bahruz Jabiyev

Abstract

Variety and complexity of cyber threats are increasing in parallel with technological advancements in computer technologies. Therefore, it is getting harder to detect and fight cyber threats. Sharing of threat intelligence is a good way to grow immunity against various and complex threats, since it enables sharing parties to learn from others' experiences and knowledge, thereby to become alert to potential threats. XML-based standards TAXII, STIX and CybOX, of which development is led by MITRE, allow us to describe cyber threats in an extensive and standardized manner and to share them effectively. The aim of this thesis is to convert STIX/CybOX formatted threat intelligence data to Suricata rules which can be readily implemented on Suricata IDS/IPS. To create rules for a Suricata engine based on latest threat information on a regular basis, will enable it to get familiar with emerging threats.

Keywords: TAXII, STIX, CybOX, Suricata

Tezin Türkçe Başlığı

Bahruz Jabiyev

Öz

Siber tehditlerin çeşit ve karmaşıklığı bilgisayar teknolojilerindeki gelişimlere paralel olarak artıyor. Bu yüzden, siber tehditleri tespit etmek ve onlarla savaşmak zorlaşıyor. Siber istihbaratın paylaşılması, çeşitli ve karmaşık tehditlere karşı korumayı artırmanın iyi bir yöntemidir, çünkü bu şekilde başkalarının bilgi ve tecrübesi kullanılarak tehditlere karşı daha uyanık olunabilir. MITRE'nin gelişimine öncülük ettiği TAXII, STIX ve CybOX standartları bize siber istihbaratın kapsamlı ve standartlaştırılmış bir şekilde tarif edilmesini ve etkin olarak paylaşılmasını sağlıyor. Bu tezin amacı, STIX/CybOX formatındaki siber tehdit istihbaratını, kolayca Suricata IPS/IDS sistemlerinde tanımlanabilecek Suricata kurallarına dönüştürmek. Düzenli olarak, güncel siber tehditleri baz alarak oluşturulan Suricata kuralları, Suricata sisteminin yeni yaranan tehditlere aşina olmasına olanak sağlayacaktır.

Anahtar Sözcükler: TAXII, STIX, CybOX, Suricata

First, I want to dedicate my work to my dear future wife, Günay. She supported me through the period of my thesis writing and shared my happiness during the times I was starting, writing and getting closer to the end of my thesis. Also, I want to dedicate this work to my family, who have brought me up and who have been my lovely and kind supporters in every phase of my life.

Acknowledgments

I would like to thank to my kind and polite instructor Necati Ersen Şişeci, who tirelessly helped me to write this thesis, despite the fact that I was interrupting him often to ask my questions. I also feel myself grateful to Assistant Professor Hatice Tekiner Moğulkoç and Professor Erkan Ture who helped me a lot to start to this program.

Contents

Declaration of Authorship ii

Abstract iii

Öz iv

Acknowledgments vi

List of Figures viii

Abbreviations ix

1 Cyber Dimension of Life 1
- 1.1 Benefits of Computer Technologies . . . 1
- 1.2 Application Areas of Computer Technologies . . . 2

2 Cyber Threats and Countermeasures 3
- 2.1 Examples for Cyber Threats . . . 4
- 2.2 Security Measures Taken Against Cyber Threats . . . 5
  - 2.2.1 Anti-Malware Software Products . . . 5
  - 2.2.2 Preventive Network Devices . . . 5

3 Cyber Intelligence Sharing 7
- 3.1 Cyber Intelligence . . . 7
- 3.2 Importance of Sharing . . . 8
- 3.3 TAXII Protocol for CTI Sharing . . . 8
- 3.4 CybOX . . . 9
- 3.5 STIX . . . 13

4 Suricata 20

5 Related Work 23

6 Our Work 25

7 Conclusion 27

8 Future Work 29

List of Figures

3.1 Customizing a Feed Poll 19
3.2 Creating a Feed 19

Abbreviations

CTI	**C**yber **T**hreat **I**ntelligence
CybOX	**C**yber **O**bservable **E**xpression
C2	**C**ommand and **C**ontrol
DDoS	**D**istributed **D**enial of **S**ervice
DNS	**D**omain **N**ame **S**ystem
HTTP	**H**yper**T**ext **T**ransfer **P**rotocol
HUMINT	**HUM**an based **INT**elligence
ICS	**I**ndustrial **C**ontrol **S**ystem
IDS	**I**ntrusion **D**etection **S**ystem
IP	**I**nternet **P**rotocol
IPS	**I**ntrusion **P**revention **S**ystem
OSINT	**O**pen **S**ources **I**ntelligence
SIEM	**S**ecurity **I**nformation and **E**vent **M**anagement
SMB	**S**erver **M**essage **B**lock
SMTP	**S**imple **M**ail **T**ransfer **P**rotocol
STIX	**S**tructured **T**hreat **I**nformation e**X**pression
TAXII	**T**rusted **A**utomated e**X**change of **I**ndicator **I**nformation
TCP	**T**ransmission **C**ontrol **P**rotocol
URL	**U**niform **R**esource **L**ocator
XML	e**X**tensible **M**arkup **L**anguage
WAF	**W**eb **A**pplication **F**irewall

Chapter 1

Cyber Dimension of Life

After computer systems started to play a central role in every part of mankind's life, a new dimension emerged into our lives, cyber dimension. The term "cyber" is used to refer to things related to computer and internet. Computers or computerized systems span larger portion of our lives with each passing day. Computer technologies have innumerable benefits for us, but we should also be aware of threats brought into our lives by the fast expansion of them.

1.1 Benefits of Computer Technologies

The term "computer technologies" is commonly used for systems based on computer, like personal computers, tablets, smart phones, industrial control systems and so on. The most notable abilities of computers technologies are speed and accuracy. They facilitate communications over very long distances, speed up production of many things to large extent, store and manipulate large amount of data, do heavy and precise calculations and make many other things possible in various areas including science, health, critical infrastructures and large enterprise organizations. To better describe the role of computer technologies in the areas stated above, it is absolutely reasonable to say that, current level reached in those areas couldn't be reached without computer technologies.

1.2 Application Areas of Computer Technologies

Scientific researches exploit the power of computers in order to increase the accuracy and speed of calculations to the maximum level. Quantum Physics and Molecular Biology are good examples for branches of science in which computer power is indispensable. Health sector is also highly dependent on computer technologies. Almost all medical devices are computer based systems. Computers take the central role in the management of patients' medical data and facilitates doctor-patient relationship. Today, critical infrastructures also extensively use the computer technologies to achieve their aims. Industrial control systems owe its ability to do precise calculations and rapid operations to their computer-based nature. Enterprise organizations also use computers to achieve their career goals and manage their internal affairs. To mention all areas in which computer technologies are used and for what aims they are used in those areas, is out of scope of this work. But to describe better the extent of computer usage in today's world, it is possible to say that there is almost not an area remaining untouched by computer technologies.

Chapter 2

Cyber Threats and Countermeasures

Today, cyber domain is counted as fifth domain of war after land, sea, air and space because of its potential to be destructive as other domains. Mankind has witnessed a large number of devastating attacks coming from the cyber domain targeting governments, critical infrastructures, enterprise organizations and other important entities. It is not difficult to imagine how disastrous it is for a critical infrastructure to be out of service for many hours or days as a result of a cyber attack. And many large enterprise organizations are confronted with cyber attacks which are real threat to their reputation, privacy and availability. It is also possible to show many examples of cyber attacks targeting banks or other financial organizations which have resulted in theft of mind-boggling amounts of money and customer information. Governmental organizations also are under threat of espionage-motivated cyber attacks which are mostly state-sponsored and well-organized attacks. In response to the threats coming from cyber domain for which we gave a short description, hardware and software-based protection techniques have been developed and deployed over the last decades. However, experiences till today has shown us that developed security measures are not sufficient by themselves for full protection from cyber attacks.

2.1 Examples for Cyber Threats

Here are some main types of cyber threats and short descriptions for them:

Malware – Malware is a shortened name for Malicious Software. Malware is a program created to do malicious activities on the target system. These malicious activities may include taking control of the victim system and get commands from remote attacker. Malwares can also be spy programs which are logging every keystroke, recording the sound around or video in order to send it to the attacker. Some types of malware make threats against the victim and force them to do something, like encrypt all files and force them to pay to decrypt them.

Advanced Persistent Threats – APTs are mostly very well-organized, very creepy and dangerous programs. Most of the time, they are tailor-made programs for their target. This type of malware is designed to persist for long time in the victim systems and do malicious activities on them like collecting information about systems and sending it to the attacker.

Botnet – Botnet is a name given to networks made up of a large number of infected computers, sometimes hundreds of thousands. These computers are malware infected and controlled by Command and Control (C2) Servers. C2 Servers are the main actors of most Distributed Denial of Service (DDoS) attacks. They use their bots (or infected computers) to conduct attacks to a specific target by making each of them send packets to the target system, so that cause overload on the target, which generally results in denial of service. Most of the time, the motivation behind these attacks is to take down the target server and cause availability problems, which in turn may cause monetary damage, loss of reputation and so on.

Social Engineering – Social engineering attacks, also known as phishing attacks are tailor-made attacks for the victim. In these attacks, usually a trusted person or an organization is impersonated. The main goal in these attacks is to deceive the victim into downloading some malicious file or visiting some malicious site and so that gain access to the internal network or steal the credentials of the victim. Other attack scenarios are also possible.

2.2 Security Measures Taken Against Cyber Threats

In order to mitigate cyber threats, many security products have been developed to date. Some of them are fighting local threats within the host, protecting local file system against malicious files. Once they are detected they are removed from the system. Other type of taken security measures include fighting cyber attacks coming over network, by detecting threats either in transport layer or in the application layer.

2.2.1 Anti-Malware Software Products

Anti-Malware products, commonly known as antivirus programs, mainly monitor local filesystems for any illegal file or connection. Antivirus products rely mainly on a database of identifiable pieces of known suspicious code (file signatures), as well as behavioral and pattern-matching analysis (heuristics) to identify suspect files [1, p.10].

2.2.2 Preventive Network Devices

Several types of network devices have been emerging till today, each of them responsible for fighting different aspects of network intrusions. Some of them are vitally important so that it is virtually impossible for a company to function properly without them. Almost all types of network devices are getting deployed more commonly as a natural result of increasing complexity of cyber attacks that companies are exposed to. Below, some major types of network devices have been listed with short explanations about them.

Firewall – Nowadays, it is not possible to imagine a company which has a computer network inside and does not deploy a firewall which serves as an exterior gate for the company's network. Its main goal is to isolate company's internal network from outside world. Based on rules written to a firewall, it is filtering incoming and outgoing network traffic.

Web Application Firewall – As cyber attacks, targeting security holes in web applications, are increasing in popularity, Web Application Firewalls (WAF) are deployed to combat them. Fundamentally, they monitor application layer part of packets and drop them when they detect dangerous pattern in them.

Anti-DDoS Devices – As we have already mentioned, Distributed Denial of Service attacks, DDoS attacks in short, are remaining as one of the most serious threats for a company's availability, in other words, its capability to serve its customers all the time. To mitigate this sort of attacks, Anti-DDoS devices have been developed. They guard company's network by blocking traffic surpassing certain level in terms of packet rate or number of packets from a single source.

Intrusion Prevention/Detection System – In short IPS/IDS, is among the most successful security solutions which is able to analyze packets at network, transport layers and additionally at application layer for some protocols. Normally, these systems have hundreds or thousands of rules implemented on them for detecting dangerous packets. If a matched packet is found, it generates log about it. If it is deployed in IPS/inline mode, it also prevents packets from passing, besides generating log about that packets. Suricata is one of the most successful and well-known IPS/IDS engines. Since Suricata is closely related to the topic of this thesis, shortly we will return back to Suricata and give detailed information about it.

Chapter 3

Cyber Intelligence Sharing

One way to grow immunity to cyber attacks threatening us is to create a cooperation in which cooperating parties share their experiences, observations, knowledge with other parties to give them a chance to be aware, to assess and adopt, so that help them to protect themselves better. As the number and variety of parties involved and the complexity of shared information increases, there will be more need to do sharing in time and cost-efficient manner. This is possible through a standardized and comprehensive sharing protocol adopted by each cooperating party.

3.1 Cyber Intelligence

Cyber intelligence is a general term used to refer to information which is valuable from cyber security aspect. This can be in various forms, may be an IP address of C2 server which should be blocked, may be the hash of a malicious file which should be removed once detected, may be a malicious e-mail which should be regarded as a phishing attempt or may be an attack technique which should be associated with the related threat actor when observed. When cyber intelligence is shared, it will enable us to take concrete actions against related threats and protect ourselves better.

3.2 Importance of Sharing

When increasing complexity of cyber attacks is considered, Cyber Threat Intelligence (CTI) sharing is key to succeed in cyber defense. Sharing is a good first step to make a defensive cooperation. By sharing cyber threat information, each party in the cooperation can build up a stronger immunity against threats, given that cyber actors can use same techniques for more than one target. Sharing can also help to paint a more complete portrait about threat actors and techniques they use. To explain it with example, say, you have a malicious file in your hand, others' observations and knowledge about that specific file will help you to better understand the goals of this malicious file, or vice versa, after getting some information about that malware you can share it to help others to handle the same situation if they encounter it in the future. One important aspect of threat intelligence sharing is that, network products can also become more effective by sharing intelligence with each other. A security tool can automatically convert threat intelligence to something actionable. By leveraging a comprehensive and standardized threat intelligence sharing protocol, it is possible to enable time and cost-efficient communication of entities of different natures and types. To use common language for sharing intelligence is also helpful for avoiding ambiguities.

3.3 TAXII Protocol for CTI Sharing

TAXII, which is the abbreviation of "Trusted Automated Exchange of Indicator Information", is an open-source and community-driven project, which is being developed by MITRE Corporation. The main goal of TAXII project is to provide a standard way for sharing threat intelligence between the cyber entities in a speedy and secure manner, thereby establishing a fulfilling sharing environment. To carry threat data, TAXII protocol uses HTTP, which enables TAXII to take advantage of HTTP security mechanisms like encryption and authentication [2]. TAXII Protocol also works well with its native threat expression formats like CybOX and STIX of which development also led by MITRE Corporation.

3.4 CybOX

Cyber Observable Expression (CybOX) is an XML format standard to describe observed cyber entities including but not limited to:

- Action (e.g. Create File)
- Domain
- Email
- File
- IP Address
- Network Connection
- URL

Below, a CybOX expression for an action of creating a file is shown. In this action, a file, `bad_file` with MD5 hash value of `D3AA2F5A6B937885DE25302801E74AB2` is successfully created in `C:\Windows\system32`. It is quite common to observe such an action while analyzing malware samples.

```
<cybox:Action id="example:Action-b57aa65f-9598-04fb-a9d1-5094c36d5dc4" action_status="
    Success" context="Host" timestamp="2016-05-28T09:22:00.0Z">
 <cybox:Type xsi:type="cyboxVocabs:ActionTypeVocab-1.0">Create</cybox:Type>
 <cybox:Name xsi:type="cyboxVocabs:ActionNameVocab-1.0">Create File</cybox:Name>
 <cybox:Associated_Objects>
  <cybox:Associated_Object id="example:Object-043d8340-0300-46ee-b3bd-27693c8f64b7">
   <cybox:Properties xsi:type="FileObj:FileObjectType">
    <FileObj:File_Name>bad_file.exe</FileObj:File_Name>
    <FileObj:File_Path>C:\Windows\system32</FileObj:File_Path>
    <FileObj:Hashes>
     <cyboxCommon:Hash>
      <cyboxCommon:Type>MD5</cyboxCommon:Type>
      <cyboxCommon:Simple_Hash_Value datatype="hexBinary">D3AA2F5A6B937885DE25302801E74AB2</
    cyboxCommon:Simple_Hash_Value>
     </cyboxCommon:Hash>
    </FileObj:Hashes>
   </cybox:Properties>
   <cybox:Association_Type xsi:type="cyboxVocabs:ActionObjectAssociationTypeVocab-1.0">
    Affected</cybox:Association_Type>
  </cybox:Associated_Object>
```

```
 </cybox:Associated_Objects>
</cybox:Action>
```

Listing 3.1: CreateFile Action CybOX

Domain is among the most basic observables, which can be used to describe attack patterns. An example domain CybOX has been described below.

```
<cybox:Observable id="example:observable-0748fef1-f756-4aaa-b537-b996078a33dc">
 <cybox:Title>Cryptolocker Domain</cybox:Title>
 <cybox:Description>This domain is hosted by a Cryptolocker Server.</cybox:Description>
 <cybox:Object id="example:domainname-d10d61e5-1c63-4c1f-9f46-e079361e96ba">
  <cybox:Properties xsi:type="DomainNameObj:DomainNameObjectType" type="FQDN">
   <DomainNameObj:Value>server1.cryptolocker.com</DomainNameObj:Value>
  </cybox:Properties>
 </cybox:Object>
</cybox:Observable>
```

Listing 3.2: A Domain Name CybOX

Email is another type of observable which is commonly observed in phishing attacks. An example for malicious e-mail is described below giving information about its subject and sender.

```
<cybox:Observable id="example:observable-a91d488e-dddb-4ddd-95dd-d269b9169291">
 <cybox:Title>Cryptolocker Phishing Email</cybox:Title>
 <cybox:Description>Email delivered to infect cryptolocker malware</cybox:Description>
 <cybox:Object id="example:Email-9361c92d-bdc1-4a57-980f-4993cd74958a">
  <cybox:Properties xsi:type="EmailMessageObj:EmailMessageObjectType">
   <EmailMessageObj:Header>
    <EmailMessageObj:From xsi:type="AddressObj:AddressObjectType" category="e-mail">
     <AddressObj:Address_Value>viewyourbill@cryptolocker.com</AddressObj:Address_Value>
    </EmailMessageObj:From>
   <EmailMessageObj:Subject>View Your Bill</EmailMessageObj:Subject>
  </cybox:Properties>
 </cybox:Object>
</cybox:Observable>
```

Listing 3.3: An Email CybOX

CybOX description for an example file is given below. File as an observable type is a quite common and important one. They can say much about the attack technique, attack actor, aim of attack and many more.

```
<cybox:Observable id="example:observable-56d275d9-4513-40e5-9fb0-a76af88c8d69">
 <cybox:Title>Cryptolocker Encrypter File</cybox:Title>
 <cybox:Description>Encrypts all files using RSA algorithm</cybox:Description>
 <cybox:Object id="example:file-0b74a6e2-f749-4e25-a725-7eddfdc8746d">
  <cybox:Properties xsi:type="FileObj:FileObjectType">
   <FileObj:File_Name>YourBills</FileObj:File_Name>
   <FileObj:File_Path>AppData\Cryptolocker</FileObj:File_Path>
   <FileObj:File_Extension>.exe</FileObj:File_Extension>
   <FileObj:Size_In_Bytes>4655</FileObj:Size_In_Bytes>
   <FileObj:Hashes>
    <cyboxCommon:Hash>
     <cyboxCommon:Type xsi:type="cyboxVocabs:HashNameVocab-1.0">MD5</cyboxCommon:Type>
     <cyboxCommon:Simple_Hash_Value>376dff8959aac4bd03e9b929da8e0248</cyboxCommon:
     Simple_Hash_Value>
    </cyboxCommon:Hash>
   </FileObj:Hashes>
  </cybox:Properties>
 </cybox:Object>
</cybox:Observable>
```

Listing 3.4: A File CybOX

Botnet Server IP represented in CybOX format is given as an example below.

```
<cybox:Observable id="example:observable-a1afb57e-211c-41a3-9da5-e5a8e913cd22">
 <cybox:Title>Botnet Server</cybox:Title>
 <cybox:Description>This IP is owned by a Botnet Server.</cybox:Description>
 <cybox:Object id="example:address-97d0d894-a93a-49bf-b1a7-622386a7b416">
  <cybox:Properties xsi:type="AddressObj:AddressObjectType" category="ipv4-addr">
   <AddressObj:Address_Value>100.10.1.2</AddressObj:Address_Value>
  </cybox:Properties>
 </cybox:Object>
</cybox:Observable>
```

Listing 3.5: IPv4 CybOX

Here in this example [3], a TCP connection is described using source and destination IP addresses and port numbers.

```
<cybox:Object id="example:Object-13c00902-fc04-4d63-9362-29afedd50805">
 <cybox:Properties xsi:type="NetworkConnectionObj:NetworkConnectionObjectType">
  <NetworkConnectionObj:Layer3_Protocol datatype="string">IPv4</NetworkConnectionObj:
    Layer3_Protocol>
  <NetworkConnectionObj:Layer4_Protocol datatype="string">TCP</NetworkConnectionObj:
    Layer4_Protocol>
  <NetworkConnectionObj:Source_Socket_Address>
   <SocketAddressObj:IP_Address>
```

```
      <AddressObj:Address_Value>10.0.0.2</AddressObj:Address_Value>
    </SocketAddressObj:IP_Address>
    <SocketAddressObj:Port>
      <PortObj:Port_Value>4444</PortObj:Port_Value>
    </SocketAddressObj:Port>
   </NetworkConnectionObj:Source_Socket_Address>
   <NetworkConnectionObj:Destination_Socket_Address>
    <SocketAddressObj:IP_Address>
      <AddressObj:Address_Value>208.29.22.5</AddressObj:Address_Value>
    </SocketAddressObj:IP_Address>
    <SocketAddressObj:Port>
      <PortObj:Port_Value>443</PortObj:Port_Value>
    </SocketAddressObj:Port>
   </NetworkConnectionObj:Destination_Socket_Address>
  </cybox:Properties>
</cybox:Object>
```

Listing 3.6: A Network Connection CybOX

URL can also be described through CybOX. An example is shown below.

```
<cybox:Observable id="example:observable-3a30d6c9-7429-44b9-84bc-68a449a17cae">
 <cybox:Title>Malicious page</cybox:Title>
 <cybox:Description/>
 <cybox:Object id="example:uri-88a0337f-2db4-4ff1-b182-928c34d2f019">
  <cybox:Properties xsi:type="URIObj:URIObjectType" type="URL">
   <URIObj:Value>http://www.attacker.com/example.php</URIObj:Value>
  </cybox:Properties>
 </cybox:Object>
</cybox:Observable>
```

Listing 3.7: URL CybOX

On whole, CybOX format is used to describe building blocks of more complex types of threat intelligence which is mainly expressed in STIX format. Think of an example in which a phishing attack is done to a company. CybOX format can be used to describe observables in this attack, like malicious file attached to the e-mail or e-mail itself. Whereas to describe the whole phishing attack with various aspects is the responsibility of STIX.

3.5 STIX

Structured Threat Information Expression (STIX) aims to describe intrusion attempts in more detail and provide context to what is observed in order to understand cyber situation better. For this aim, various aspects of an intrusion attempt can be represented through STIX. Some of them are listed below:

- TTP
- Indicator
- Incident
- Campaign
- Exploit Target
- Threat Actor
- Course of Action

TTP is used to refer to attack techniques and tactics leveraged by an adversary to carry out cyber attacks. Botnet, malware, phishing can be shown among commonly-used TTPs. Identifying TTP used by an adversary can prove very helpful to gain better understanding about the threat actor and intended effects. TTP leveraged by Cryptolocker ransomware, is shown in details in STIX format as follows.

```
1  <stix:TTP id="example:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" timestamp="2016-05-06T20
       :16:24.142150+00:00" xsi:type='ttp:TTPType'>
2   <ttp:Title>Cryptolocker Ransomware</ttp:Title>
3   <ttp:Short_Description>Comes as an e-mail attachment and once downloaded and executed it
       encrypts all files</ttp:Short_Description>
4   <ttp:Intended_Effect timestamp="2016-05-06T20:16:24.142455+00:00">
5    <stixCommon:Value xsi:type="stixVocabs:IntendedEffectVocab-1.0">Extortion</stixCommon:
       Value>
6   </ttp:Intended_Effect>
7   <ttp:Behavior>
8    <ttp:Attack_Patterns>
9     <ttp:Attack_Pattern>
10     <ttp:Title>Phishing</ttp:Title>
11     <ttp:Description>Uses e-mail to deliver the malicious executable.</ttp:Description>
12    </ttp:Attack_Pattern>
```

```
  </ttp:Attack_Patterns>
  <ttp:Malware>
   <ttp:Malware_Instance>
    <ttp:Name>crypto.exe</ttp:Name>
    <ttp:Title>Encrypter</ttp:Title>
    <ttp:Short_Description>Once executed, encrypts all files.</ttp:Short_Description>
   </ttp:Malware_Instance>
  </ttp:Malware>
 </ttp:Behavior>
</stix:TTP>
```

Listing 3.8: TTP

Indicators are among the most commonly-shared form of threat intelligence. As its name suggests, indicators represent things that we should be alert to, since they indicate some sign of an attack. Indicators can be associated with attack actors and TTPs. For example, "if you see an IP of `10.1.0.0` (potentially a botnet server IP) in your network traffic, it can be the indication of threat actor named `Zeus` which is owner of C2 servers". Below, STIX for this example has been given. In the example below, TTP is referenced within the indicator.

```
<stix:Observables cybox_major_version="2" cybox_minor_version="1" cybox_update_version="0">
 <cybox:Observable id="example:observable-f8f0de7b-d07b-42cf-9a92-85e3838b7907">
  <cybox:Title>Cryptolocker Encrypter File</cybox:Title>
  <cybox:Description>Encrypts all files using RSA algorithm</cybox:Description>
  <cybox:Object id="Biznet:file-0b74a6e2-f749-4e25-a725-7eddfdc8746d">
   <cybox:Properties xsi:type="FileObj:FileObjectType">
    <FileObj:Hashes>
     <cyboxCommon:Hash>
      <cyboxCommon:Type xsi:type="cyboxVocabs:HashNameVocab-1.0">MD5</cyboxCommon:Type>
      <cyboxCommon:Simple_Hash_Value>376dff8959aac4bd03e9b929da8e0248</cyboxCommon:
    Simple_Hash_Value>
     </cyboxCommon:Hash>
    </FileObj:Hashes>
   </cybox:Properties>
  </cybox:Object>
 </cybox:Observable>
</stix:Observables>
<stix:Indicators>
 <stix:Indicator id="example:indicator-5a0c0420-e500-41f8-baa7-dd1628acb526" timestamp="
    2016-05-06T21:14:53.290760+00:00" xsi:type='indicator:IndicatorType'>
  <indicator:Title>Cryptolocker</indicator:Title>
  <indicator:Type xsi:type="stixVocabs:IndicatorTypeVocab-1.1">File Hash Watchlist</
    indicator:Type>
  <indicator:Observable idref="example:observable-f8f0de7b-d07b-42cf-9a92-85e3838b7907"></
    indicator:Observable>
```

```
22    <indicator:Indicated_TTP>
23     <stixCommon:TTP idref="example:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" xsi:type='ttp:
       TTPType'/>
24    </indicator:Indicated_TTP>
25   </stix:Indicator>
26  </stix:Indicators>
27  <stix:TTPs>
28   <stix:TTP id="example:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" timestamp="2016-05-06T20
       :16:24.142150+00:00" xsi:type='ttp:TTPType'>
29   <ttp:Title>Cryptolocker Ransomware</ttp:Title>
30   <ttp:Short_Description>Comes as e-mail attachment and once downloaded and executed it
       encrypts all files</ttp:Short_Description>
31   </stix:TTP>
32  </stix:TTPs>
```

Listing 3.9: Indicator

While indicators are serving to avoid attacks in the future, incidents represent things that already happened. In the following example an incident has been described in which Cryptolocker has caused disruption of operations after encrypting critical files.

```
1   <stix:TTPs>
2    <stix:TTP id="example:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" timestamp="2016-05-06T20
       :16:24.142150+00:00" xsi:type='ttp:TTPType'>
3    <ttp:Title>Cryptolocker Ransomware</ttp:Title>
4    <ttp:Short_Description>Comes as e-mail attachment and once downloaded and executed it
       encrypts all files</ttp:Short_Description>
5    </stix:TTP>
6   </stix:TTPs>
7   <stix:Incidents>
8    <stix:Incident id="Biznet:incident-b0f53c03-8a25-4ca4-a48e-9fe256214732" timestamp="
       2016-05-06T23:07:24.780423+00:00" xsi:type='incident:IncidentType'>
9    <incident:Title>Cryptolocker Infection</incident:Title>
10   <incident:Short_Description>Cryptolocker Encrypted Critical Files.</incident:
       Short_Description>
11   <incident:Impact_Assessment>
12    <incident:Effects>
13     <incident:Effect xsi:type="stixVocabs:IncidentEffectVocab-1.0">Disruption of Service /
       Operations</incident:Effect>
14    </incident:Effects>
15   </incident:Impact_Assessment>
16   <incident:Leveraged_TTPs>
17    <incident:Leveraged_TTP>
18     <stixCommon:TTP idref="Biznet:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" xsi:type='ttp:
       TTPType'/>
19    </incident:Leveraged_TTP>
20   </incident:Leveraged_TTPs>
21   </stix:Incident>
```

```
</stix:Incidents>
```

Listing 3.10: Incident

Type of activities targeting a particular sector with a particular motivation (e.g. stealing financial data) is called a campaign. In the example shown below, a campaign and the TTP used within campaign, related incidents and the threat actor behind the campaign has been described.

```
<stix:Campaigns>
 <stix:Campaign id="example:campaign-04999dc8-a249-4e7a-a423-31721a4a7996" timestamp="
    2016-05-06T23:21:58.767753+00:00" xsi:type='campaign:CampaignType'>
  <campaign:Title>Cryptolocker Business</campaign:Title>
  <campaign:Intended_Effect timestamp="2016-05-06T23:21:58.767900+00:00">
   <stixCommon:Value xsi:type="stixVocabs:IntendedEffectVocab-1.0">Advantage - Economic</
    stixCommon:Value>
  </campaign:Intended_Effect>
  <campaign:Related_TTPs>
   <campaign:Related_TTP>
    <stixCommon:TTP idref="example:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" xsi:type='ttp:
    TTPType'/>
   </campaign:Related_TTP>
  </campaign:Related_TTPs>
  <campaign:Related_Incidents>
   <campaign:Related_Incident>
    <stixCommon:Incident idref="example:incident-b0f53c03-8a25-4ca4-a48e-9fe256214732" xsi:
    type='incident:IncidentType'/>
   </campaign:Related_Incident>
   <campaign:Related_Incident>
    <stixCommon:Incident idref="example:incident-ea153c03-3c25-4ca4-a48e-9fe378212333" xsi:
    type='incident:IncidentType'/>
   </campaign:Related_Incident>
  </campaign:Related_Incidents>
  <campaign:Attribution>
   <campaign:Attributed_Threat_Actor>
    <stixCommon:Threat_Actor idref="example:threatactor-5d383661-e733-444f-aec1-8bed93390f84
    " xsi:type='ta:ThreatActorType'/>
   </campaign:Attributed_Threat_Actor>
  </campaign:Attribution>
 </stix:Campaign>
</stix:Campaigns>
<stix:Threat_Actors>
 <stix:Threat_Actor id="example:threatactor-5d383661-e733-444f-aec1-8bed93390f84" timestamp=
    "2016-05-06T23:21:01.776435+00:00" xsi:type='ta:ThreatActorType'>
  <ta:Title>Bad Group</ta:Title>
  <ta:Short_Description>They extort money from victims.</ta:Short_Description>
  <ta:Type timestamp="2016-05-06T23:21:01.776611+00:00">
```

```
32     <stixCommon:Value xsi:type="stixVocabs:ThreatActorTypeVocab-1.0">eCrime Actor - Organized
         Crime Actor</stixCommon:Value>
33    </ta:Type>
34    <ta:Motivation timestamp="2016-05-06T23:21:01.776752+00:00">
35     <stixCommon:Value xsi:type="stixVocabs:MotivationVocab-1.1">Financial or Economic</
         stixCommon:Value>
36    </ta:Motivation>
37   </stix:Threat_Actor>
38 </stix:Threat_Actors>
```

LISTING 3.11: Campaign

Exploit Targets are used to represent security holes (e.g. misconfigurations, vulnerabilities) that are exposed to cyber threats. In our example [**?**], a vulnerability in Mozilla Firefox has been described.

```
1 <stixCommon:Exploit_Target xsi:type="et:ExploitTargetType" id="example:et-48a276f7-a8d7-bba2
      -3575-e8a63fcd488" timestamp="2016-05-20T11:00:00.000000Z">
2  <et:Title>Buffer Overflow vulnerability in Mozilla Firefox</et:Title>
3  <et:Vulnerability>
4   <et:CVE_ID>CVE-2016-1935</et:CVE_ID>
5  </et:Vulnerability>
6 </stixCommon:Exploit_Target>
```

LISTING 3.12: Exploit Target

STIX is also able to represent threat actors. Threat actors are people behind cyber crimes. It is highly likely that, after threat actor is identified, most of other aspects of a cyber crime comes to light. Below, an example representation of a threat actor is shown.

```
1 <stix:Threat_Actor id="example:threatactor-5d383661-e733-444f-aec1-8bed93390f84" timestamp="
      2016-05-06T23:33:10.349984+00:00" xsi:type='ta:ThreatActorType'>
2  <ta:Title>Bad Group</ta:Title>
3  <ta:Short_Description>They extort money from victims.</ta:Short_Description>
4  <ta:Type timestamp="2016-05-06T23:33:10.347153+00:00">
5   <stixCommon:Value xsi:type="stixVocabs:ThreatActorTypeVocab-1.0">eCrime Actor - Organized
      Crime Actor</stixCommon:Value>
6  </ta:Type>
7  <ta:Motivation timestamp="2016-05-06T23:33:10.347322+00:00">
8   <stixCommon:Value xsi:type="stixVocabs:MotivationVocab-1.1">Financial or Economic</
      stixCommon:Value>
9  </ta:Motivation>
10 <ta:Observed_TTPs>
11  <ta:Observed_TTP>
12   <stixCommon:TTP idref="example:ttp-3b6871bc-f1b7-42b4-be83-9221f602575d" xsi:type='ttp:
      TTPType'/>
```

```
13    </ta:Observed_TTP>
14   </ta:Observed_TTPs>
15  </stix:Threat_Actor>
```

Listing 3.13: Threat Actor

As its name suggests, course of action is a set of defensive actions done either during an attack or after an attack is completed. Example below describes a course of action which aims mitigation of a bot infection and some other information about it. (e.g. cost and efficacy of action)

```
1  <stix:Course_Of_Action id="example:coa-180819dc-be19-4273-9210-42a102ae3820" timestamp="
       2016-05-06T23:42:50.296472+00:00" xsi:type='coa:CourseOfActionType'>
2   <coa:Title>Block Botnet Traffic</coa:Title>
3   <coa:Stage xsi:type="stixVocabs:COAStageVocab-1.0">Response</coa:Stage>
4   <coa:Short_Description>Stop Communication Between Internal Network and Botnet Servers</coa:
       Short_Description>
5   <coa:Impact timestamp="2016-05-06T23:42:50.296668+00:00">
6    <stixCommon:Value xsi:type="stixVocabs:HighMediumLowVocab-1.0">Low</stixCommon:Value>
7    <stixCommon:Description>Because blocked IPs are not operationally important, impact is low
       .</stixCommon:Description>
8   </coa:Impact>
9   <coa:Cost timestamp="2016-05-06T23:42:50.296620+00:00">
10   <stixCommon:Value xsi:type="stixVocabs:HighMediumLowVocab-1.0">Low</stixCommon:Value>
11   <stixCommon:Description>It costs to write a new firewall rule to block related IP.</
       stixCommon:Description>
12  </coa:Cost>
13  <coa:Efficacy timestamp="2016-05-06T23:42:50.296719+00:00">
14   <stixCommon:Value xsi:type="stixVocabs:HighMediumLowVocab-1.0">High</stixCommon:Value>
15   <stixCommon:Description>It will stop communication with Botnet Servers.</stixCommon:
       Description>
16  </coa:Efficacy>
17 </stix:Course_Of_Action>
```

Listing 3.14: Course of Action

`Soltra Edge` is a free TAXII Server, that can be installed as a virtual machine, which has been made available by Soltra[1]. With its user-friendly web interface (Figure 3.1), this server can be used to poll security feeds regularly from TAXII servers that produce security feeds. There are some freely available feed services including `http://hailataxii.com/taxii-discovery-service` and `http://edge.threatactorlab.com/taxii-discovery-service`, that are producing threat intelligence and sharing them with clients.

[1]Soltra is a FS-ISAC and DTCC company.

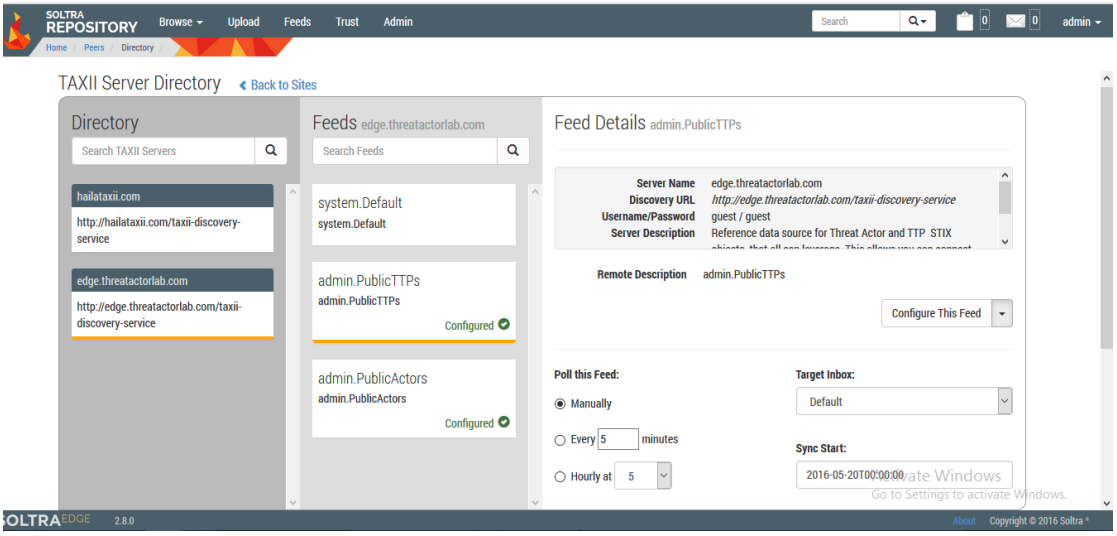

FIGURE 3.1: Customizing a Feed Poll

`Soltra Edge` also enables producing cyber intelligence and sharing it with others. In fact, the web interface of Soltra Edge makes the job of creating feeds in STIX/CybOX format much easier (Figure 3.2). Once a created feed is saved and published, it is ready to be consumed by others.

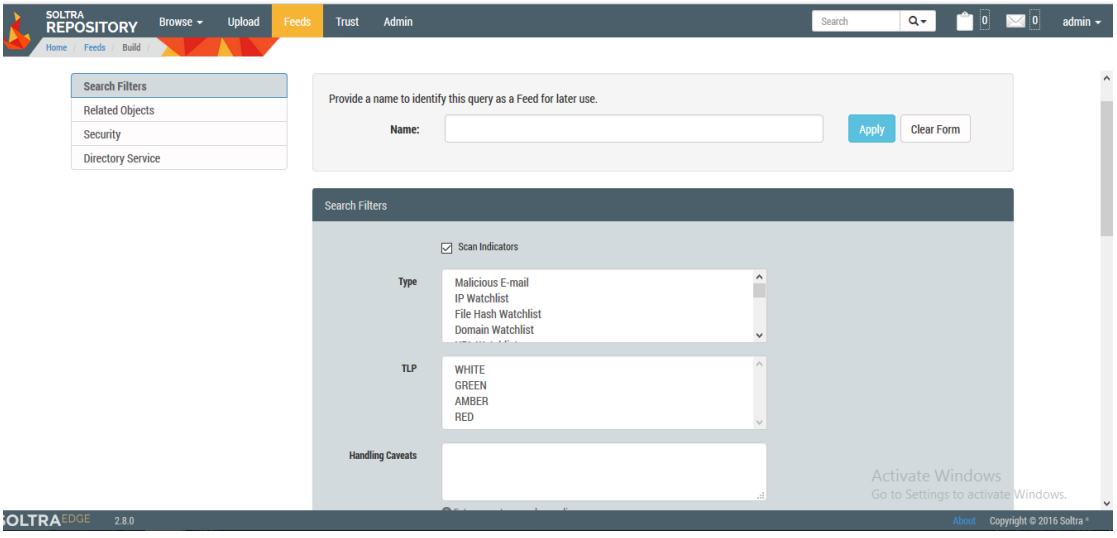

FIGURE 3.2: Creating a Feed

Chapter 4

Suricata

Suricata is a high performance Network IDS, IPS and Network Security Monitoring engine. Open Source and owned by a community run non-profit foundation, the Open Information Security Foundation (OISF). Suricata is developed by the OISF and its supporting vendors [4]. Based on rules implemented on Suricata it is able to analyze packets at network, transport layers and additionally at application layer for some protocols including HTTP, DNS, SMB and more, and decides whether they are dangerous packets or not, in the case of a packet is dangerous, it generates log or blocks it or do both depending on the mode Suricata works in. A simple Suricata rule is given as an example below.

```
drop tcp $HOME_NET any -> 10.10.10.10 6667
(msg:"Poison Ivy Trojan Detected."; content:"NICK";
reference:url,example.com/bot_servers; sid:123456; rev:2;)
```

First keyword in the rule above, namely `drop` defines the action to take when this signature matches, as its name suggests a packet with matched signature will be dropped. There are some additional action keywords. They are `pass`, `reject`, `alert`. Names of actions are quite self-explanatory. The second keyword in the rule above, `tcp` shows the protocol which this signature concerns. `$HOME_NET` is a name given to some IP interval (e.g. 10.1.0.0/24) which is defined in Suricata.yaml, Suricata's configuration file. It describes source address. Because `any` is given as source port, all ports hold for this

signature to match. The destination IP of a packet must be `10.10.10.10` and the destination port `6667` to match this signature. Message part, which is `msg:"..."` gives more information about the signature and the possible alert [5]. To look for a pattern in a packet `content` keyword is used, in this case `NICK` is a word of interest. To give the reference for a signature, from where you can get information about the signature, `reference` keyword is used. To identify signatures , a unique number, which is called `sid` is given to each signature. Lastly, `rev` is for revision tracking, which increases by one, every time the author makes a change to the rule. On whole, this signature tries to match patterns of malicious C2 packets, once signature matches, packet is dropped and is not allowed to go any further.

HTTP Keywords – HyperText Transfer Protocol (HTTP) is one of layer 7 protocols at which Suricata is able to inspect packets. Suricata makes it possible to write rules based on all parts of HTTP payload, from HTTP method to HTTP body, using HTTP keywords. These include `http_method`, `http_uri`, `http_header`, `http_cookie`, `http_user_agent`, `http_client_body` and several more. An example, including usage of both `http_uri` and `http_header`, is given below.

```
drop http $HOME_NET any -> $EXTERNAL_NET 80
(msg:"Malicious site visit"; content:"www.attacker.com";http_header;
content:"/example.php";http_uri; sid:1234567;)
```

If a packet visits `http://www.attacker.com/example.php`, this rule will drop that packet and generate an alert for it.

DNS Keywords – DNS (Domain Name System) is a protocol for domain name resolution. Every time a user visits a website, first, website address (e.g. www.google.com) is resolved over DNS protocol, then it becomes possible to go to that address. Using DNS keywords it is possible to inspect and filter DNS packets. Example below, shows the usage of `dns_query`, a keyword that is able to inspect DNS packets.

```
alert dns any any -> any 53 (msg:"Malicious resolution attempt";
dns_query; content:"www.malicious.com"; nocase; sid:54321;)
```

This rule will generate an alert every time it detects the word `www.malicious.com` in a DNS response.

File Keywords – Starting with Suricata version 1.2 it's possible to extract files from HTTP sessions as well as match on file name, extension and "magic" [6]. Also MD5 hash of files downloaded, can be calculated on the fly. Combining the hash feature and features as `fileext`, `filemagic`, `filename`, there is a great chance to identify threatening files crossing our network. What's more, it is also possible to store files to the disk in order to analyze them further. Following rule will calculate hashes of windows executable files on the fly and compare them against hashes in `BadMD5s.txt`, in case of match, an alert will be generated for that matched file.

```
alert http any any -> any any (msg:"File MD5 Hash Matches.";
filemagic:"exe"; filemd5:BadMD5s.txt; sid:12345;)
```

Chapter 5

Related Work

In order to explore the ways of getting most out of cyber intelligence sharing, we have made use of several articles on this topic. While some of articles describe real-world applications of STIX and CybOX technologies, some are seeking methods for effective use of collected threat intelligence.

In [7], authors have described how a new generation SIEM is being deployed in ACDC (Advanced Cyber Defence Centre) project, to fight botnets and other cyber threats, correlating local security logs and normalized STIX data provided by ACDC partners. Correlation directives defined on the SIEM enables it to identify what is important from enormous amount of events collected. These include quantitative rules (e.g. has specific IP address been sighted three times), temporal rules (e.g. has specific file hash been observed twice within a week) and qualitative rules (e.g. has specific URL address been reported by three different tools). Based on these rules, observations that have reached sufficient level of reliability is transmitted to CCH (Centralized Data Clearing House) which in turn shares those with other parties. The new generation SIEM deployed in ACDC project becomes more effective in fighting cyber threats after combining STIX events provided by external tools and local security logs, besides producing and sharing cyber intelligence extracted from the combination of two.

Authors in [8] suggest a way to extract specific information from huge amounts of observations captured by network tools and host based applications with the help of inference engines. For that aim, they describe an information relevance reasoning mechanism. To understand a cyber situation better, several queries (e.g. Which servers are targeted?) are made about it. To answer that queries about a situation, inference engines should

come into play, which use ontology languages to make inferences automatically. In that project, authors use BaseVISor as inference engine and STIX Ontology which uses STIX components like (TTPs, Indicators etc.) as its classes.

In [9], the author argues that, when compared to enterprise ecosystem, cyber attacks made to Industrial Control Systems (ICS) are harder to detect and combat because of the different nature of control systems and limited amount of resources available. To develop understanding to ICS environment attacks and machine-speed remediation to them, MMATR (Machine-to-Machine Automated Threat Response) project started in 2013. This project consists of three parts: 1) enriching CybOX with some additional features like adding temporal context, 2) developing remediation in machine-readable and machine-actionable format, 3) developing machine-readable specifications containing threat information and mitigation steps.

It is proposed in [10] to combine data collected from sources in the military and civil environment, namely traditional intelligence with technical data collected (e.g. file hashes, IP addresses, TTPs) to gain a better understanding about a cyber threat. HUMINT (Human Based Intelligence) and OSINT (Open Sources Intelligence) are among the given examples for available sources of conventional information from which it is possible to get information about names, locations, motivations, IP addresses, homepages, blogs and etc. Other resources suggested in the article include social media, chat rooms, hacker forums which may prove very useful to paint a clearer situational picture.

Chapter 6

Our Work

STIX and CybOX aim to represent threat intelligence in a manner that is extensive and easily understandable. While one option is to share threat intelligence with humans so that they make use of it, another option is to share it with network devices so that they automatically learn them. However, in some cases it is only possible through converting threat intelligence to a machine-readable format. For this reason, we have written a script. This code connects to a TAXII server, download STIX/CybOX formatted feeds available and generate Suricata rules based on the data available in feeds. Generated Suricata rules are ready to be implemented on a Suricata engine. You can download this script from `https://github.com/behruzcebiyev/ConvertSTIXtoSuricataRules`.

This script is able to extract IP addresses in a feed and make a rule at which a list of IP addresses is matched. Every time a feed is polled and IP addresses are extracted, that list of IP addresses, namely the blacklist will grow. This rule will enable Suricata to recognize emerging bad IPs and alert or drop them once detected. Likely, malicious domains can be extracted from a feed with the help of the script. After extracting domain names, the script will create DNS level rules for Suricata which will look for that domains in DNS responses. If a DNS response contains such a domain name, it is an indication of a client in our network, attempting to go to that address and awaiting name resolution to happen. This script will help us to be alert to previously unknown malicious domains visited by users in our local network.

Another type of threat that an IPS/IDS engine is supposed to be fighting against, are malicious pages. This script is also able to generate Suricata rules at HTTP level using HTTP keywords after extracting URL data from a feed downloaded.

Considering the Suricata feature of calculating the hash of a file opened/downloaded over HTTP and matching that hash against blacklist of hashes, another feature that we have added to the script is to extract MD5 hash values from a feed and update the MD5 hash blacklist with newest threats' hash values.

Initially , we wanted to add to our script the feature of generating Suricata rules at SMTP level based on mail subject, mail server address. Afterwards, we thought that Suricata rules would be insufficient to detect illicit SMTP traffic based on these fields, given that attackers use e-mails with tons of different subjects and addresses and implementing such rules on Suricata would affect its performance badly. That's why we changed our mind.

On whole, our work aims to pull STIX/CybOX formatted cyber intelligence from a TAXII server (like hailataxii.com) and convert it to a readable and actionable format by Suricata IDS/IPS engine, namely generate Suricata rules which can be readily implemented on a Suricate engine. This, in turn, will keep the Suricata engine up-to-date and make it able to defend local networks against latest cyber threats.

Chapter 7

Conclusion

TAXII, STIX and CybOX are among the most successful and the most recent initiatives to meet the rapidly growing need to share cyber threat intelligence smoothly and effectively. To enable network devices to convert delivered cyber threat intelligence into something actionable is another major need, as it is obvious that machines with this ability will be more successful in fighting emerging threats. As the practical part of our thesis we developed a python script for polling a feed and extracting specific data (like IP address, file hash and etc.) out of the feed and generating Suricata rules based on extracted data. Basically what it does is to get a threat intelligence feed and to convert it to Suricata IDS/IPS rules. This script extracts several types of data out of a feed and generate rules based on them. They are listed below.

- IP address
- Domain name
- URL
- File MD5 hash

During tests, our script worked as expected. It properly generated wanted rules and files which is to be referenced at a rule (e.g. BlackMD5s.txt is referenced at the MD5 blacklisting rule). It can also be customized to make polls on a regular basis, polling packages only available after the last poll time. This ability of the script also tested by

setting it as a cron job making polls every hour. It also worked successfully. However, further features can be added to the script to expand its usage and scope.

Chapter 8

Future Work

Currently, our script generates Suricata rules for 4 cases.

- bad IP address
- malicious domains
- malicious URLs
- bad files based on their MD5 hashes

It would be very useful for the security community, if we published that Suricata rules on the web, since web is an invaluable place to reach others and share that rules, mostly created based on recent security events. So, this can be a good next step following this project. Support for regular expressions in the processes of extracting and creating rules can be another useful feature to add. Also in the future, support for additional cases and other protocols can be added to the script. Also, since rule formats of IPS/IDS engines are similar, the output of this script can be adapted to other engines as well, like Snort, Bro IDS and etc.

Bibliography

[1] M. Sikorski and A. Honig. *Practical Malware Analysis : The Hands-On Guide to Dissecting Malicious Software*. No Starch Press, San Francisco, CA, USA, 2012.

[2] Stop sending me threat intelligence in email. http://www.tripwire.com/state-of-security/security-data-protection/cyber-security/stop-sending-me-threat-intelligence-in-e-mail/. Accessed: 2016-05-26.

[3] Cybox_network_connection_http_instance.xml. https://github.com/CybOXProject/schemas/blob/master/samples/CybOX_Network_Connection_HTTP_Instance.xml. Accessed: 2016-06-05.

[4] Suricata. https://suricata-ids.org/. Accessed: 2016-05-29.

[5] Meta-settings. https://redmine.openinfosecfoundation.org/projects/suricata/wiki/Meta-settings, . Accessed: 2016-05-29.

[6] File extraction. https://redmine.openinfosecfoundation.org/projects/suricata/wiki/File_Extraction, . Accessed: 2016-06-01.

[7] B. G. N. Crespo and A. Garwood. Fighting botnets with cyber-security analytics: Dealing with heterogeneous cyber-security information in new generation siems. In *Availability, Reliability and Security (ARES), 2014 Ninth International Conference on*, pages 192–198, Sept 2014. doi: 10.1109/ARES.2014.33.

[8] S. Lu and M. M. Kokar. A situation assessment framework for cyber security information relevance reasoning. In *Information Fusion (Fusion), 2015 18th International Conference on*, pages 1459–1466, July 2015.

[9] D. Rhoades. Machine actionable indicators of compromise. In *2014 International Carnahan Conference on Security Technology (ICCST)*, pages 1–5, Oct 2014. doi: 10.1109/CCST.2014.6987016.

[10] A. Kornmaier and F. Jaouän. Beyond technical data - a more comprehensive situational awareness fed by available intelligence information. In *Cyber Conflict (CyCon 2014), 2014 6th International Conference On*, pages 139–154, June 2014. doi: 10.1109/CYCON.2014.6916400.