ZAM

zeek asset manager

Design Review 3

Akiel Aries - Release Manager
Cody Beck - Team Architect
David Knight - Team Lead
Nathan Chan - Front-End Engineer

Team Mentor: Vahid Nikoonejad Fard
Instructor: Michael Leverington

Network Security

Crucial to the precautionary actions taken by organizations

Since 2013 there are 3,809,448 records stolen from breaches every day. 158,727 per hour, 2,645 per minute and 44 every second of every day reports Cybersecurity Ventures.

What is Zeek?

- Network traffic analyzer
- Network security monitor
- Open Source

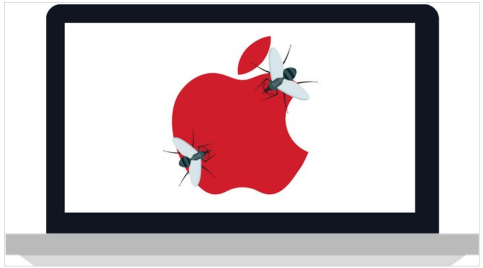
Meet Tim!

Zeek Package Manager

- Create and publish packages
- Allows users unrestricted access
- Tools that enhance network analysis

Current package website logo

Catching FruitFly



Discovery

In 2017, researchers from the Berkeley Lab using Zeek noticed strange AFP traffic on their computers

Searching

Easily searched through 2 years of logs using Zeek

Analyzing

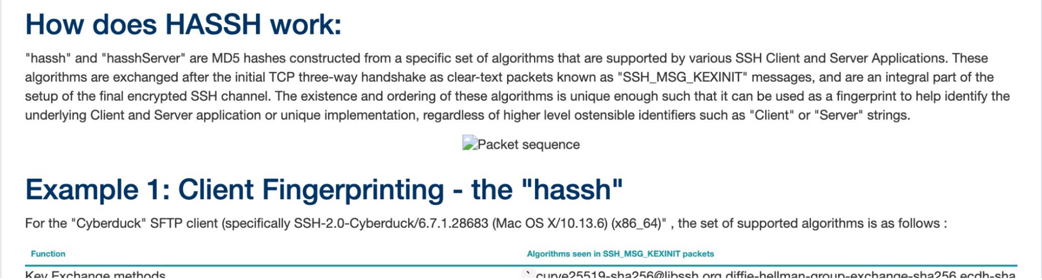
Found strange IoC’s connected to a university in Ohio

Uncovering

Emailed said university and called the FBI which led to the discovery of the Fruitfly malware

Problem Statement

- Current search feature is substandard
- Irrelevant/Incomplete results
- Markdown not rendering
- Outdated design



How does HASSH work:

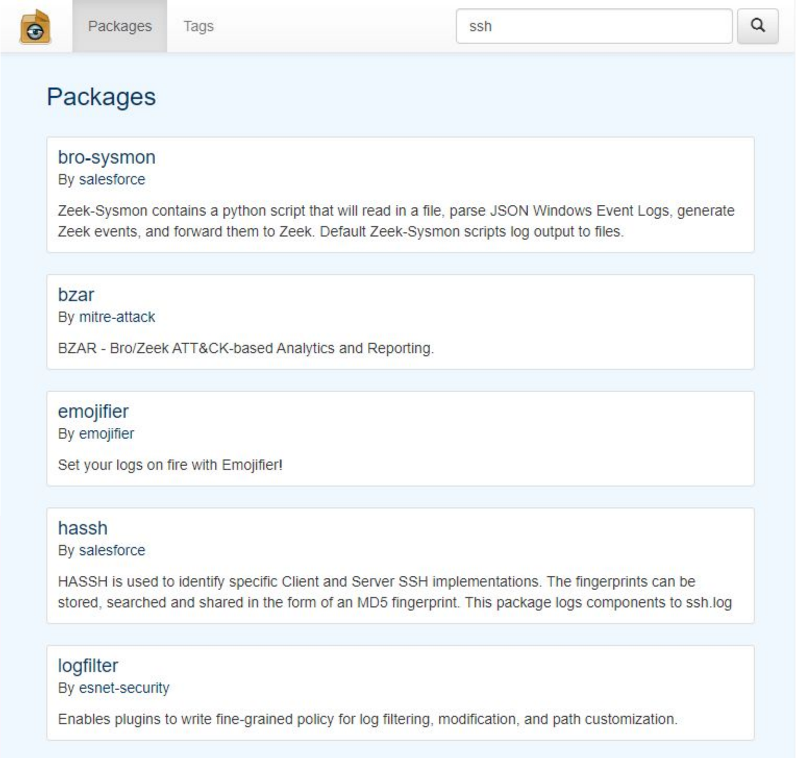
"hassh" and "hasshServer" are MD5 hashes constructed from a specific set of algorithms that are supported by various SSH Client and Server Applications. These algorithms are exchanged after the initial TCP three-way handshake as clear-text packets known as "SSH_MSG_KEXINIT" messages, and are an integral part of the setup of the final encrypted SSH channel. The existence and ordering of these algorithms is unique enough such that it can be used as a fingerprint to help identify the underlying Client and Server application or unique implementation, regardless of higher level ostensible identifiers such as "Client" or "Server" strings.

Packet sequence

Example 1: Client Fingerprinting - the "hassh"

For the "Cyberduck" SFTP client (specifically SSH-2.0-Cyberduck/6.7.1.28683 (Mac OS X/10.13.6) (x86_64)" , the set of supported algorithms is as follows :

Function	Algorithms seen in SSH_MSG_KEXINIT packets
Key Exchange methods	curve25519-sha256@libssh.org,diffie-hellman-group-exchange-sha256,ecdh-sha



Packages

bro-sysmon
By salesforce

Zeek-Sysmon contains a python script that will read in a file, parse JSON Windows Event Logs, generate Zeek events, and forward them to Zeek. Default Zeek-Sysmon scripts log output to files.

bzar
By mitre-attack

BZAR - Bro/Zeek ATT&CK-based Analytics and Reporting.

emojifier
By emojifier

Set your logs on fire with Emojifier!

hassh
By salesforce

HASSH is used to identify specific Client and Server SSH implementations. The fingerprints can be stored, searched and shared in the form of an MD5 fingerprint. This package logs components to ssh.log

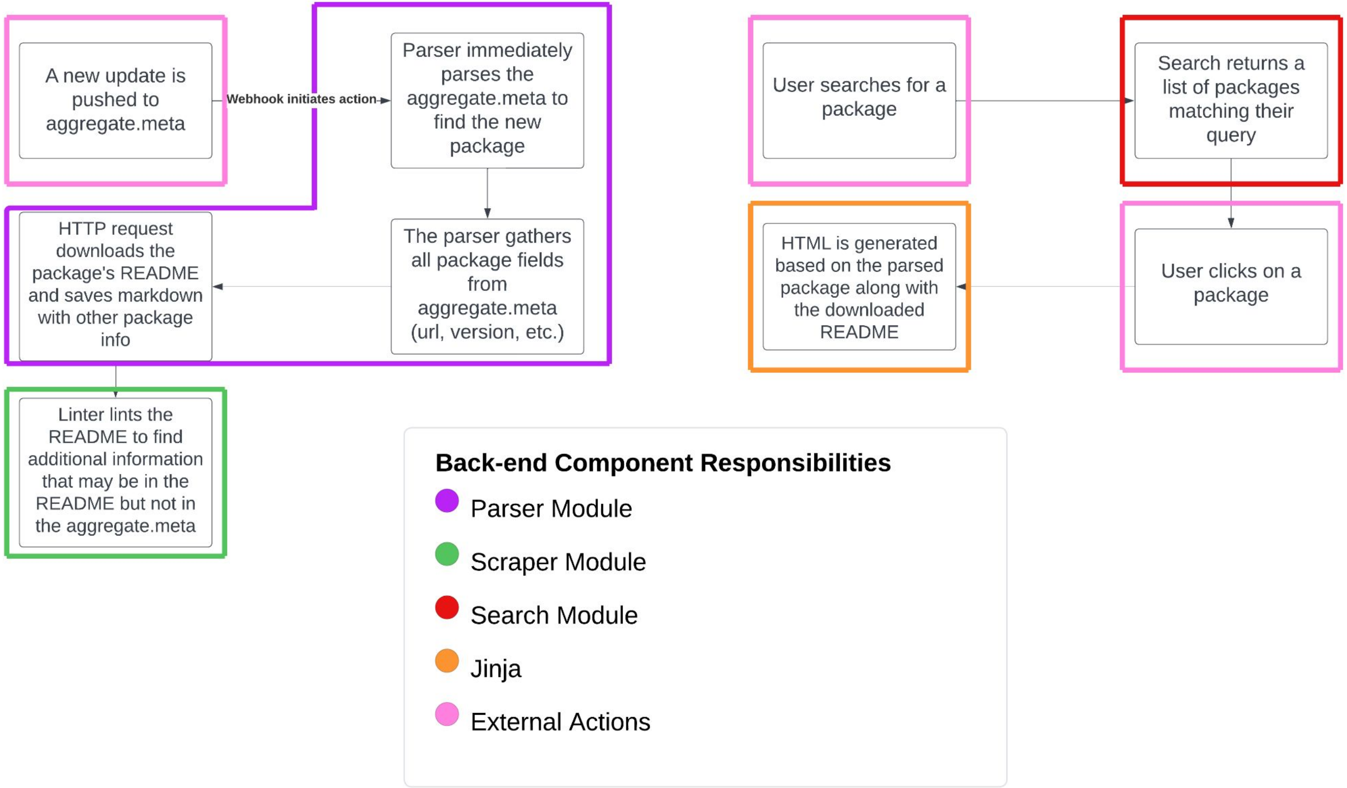
logfilter
By esnet-security

Enables plugins to write fine-grained policy for log filtering, modification, and path customization.

Key Requirements

- Parser that can get information from a central metadata file containing information about every Zeek package
- Pull additional package information, such as README files, from each package’s GitHub repository
- Scraping of packages at parse-time to highlight important information about each one
- Improved search functionality to make packages more discoverable
- A modern front-end preserving the look and feel of the zeek.org site
- Correctly render markdown from *README* files

Solution Overview



A new update is pushed to aggregate.meta

Webhook initiates action

Parser immediately parses the aggregate.meta to find the new package

The parser gathers all package fields from aggregate.meta (url, version, etc.)

HTTP request downloads the package's README and saves markdown with other package info

Linter lints the README to find additional information that may be in the README but not in the aggregate.meta

User searches for a package

Search returns a list of packages matching their query

User clicks on a package

HTML is generated based on the parsed package along with the downloaded README

Back-end Component Responsibilities

● Parser Module

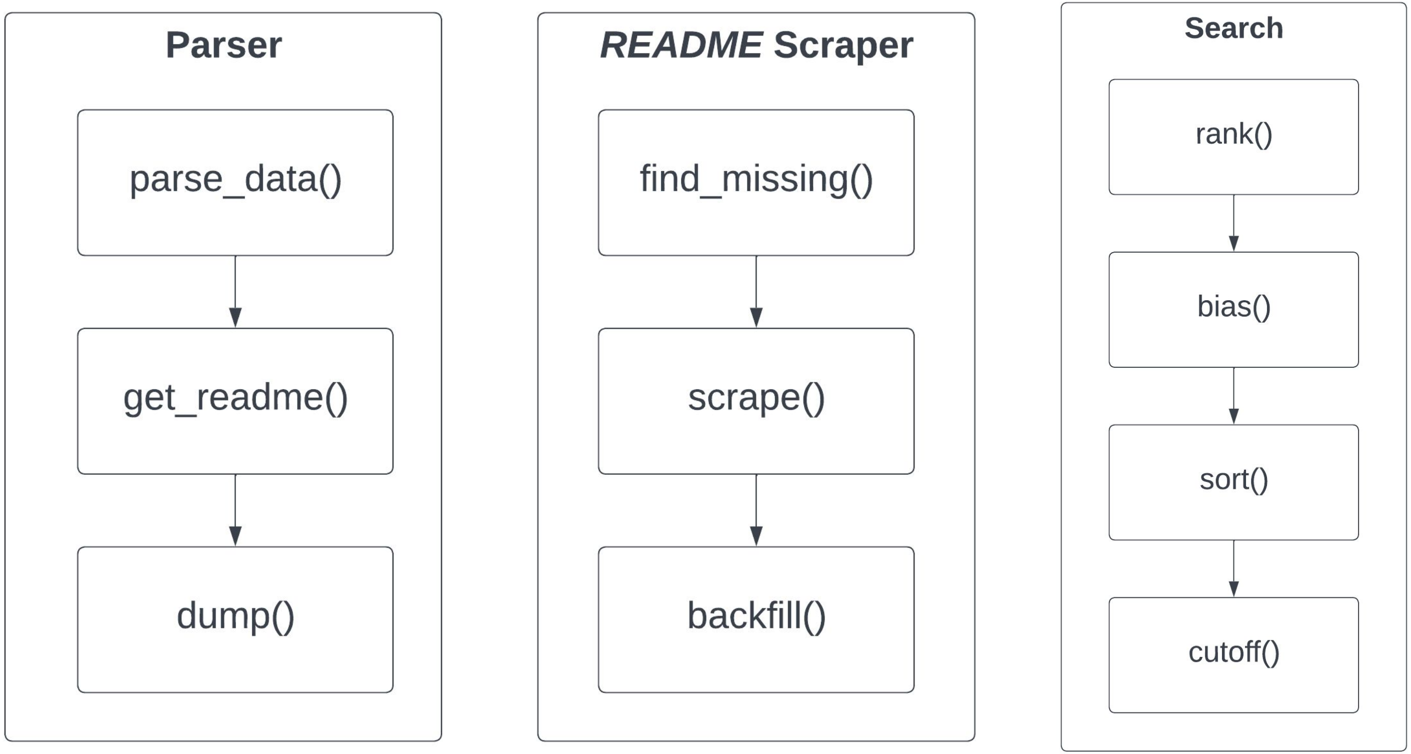
● Scraper Module

● Search Module

● Jinja

● External Actions

Architecture



Parser

parse_data()

get_readme()

dump()

README Scraper

find_missing()

scrape()

backfill()

Search

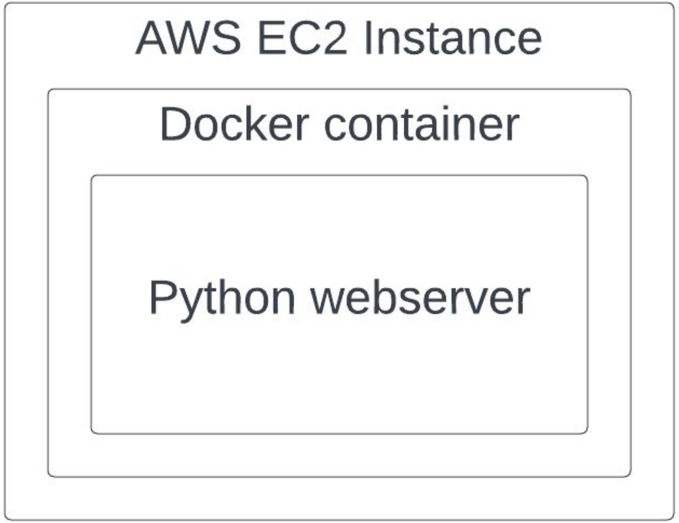
rank()

bias()

sort()

cutoff()

Architecture Continued



AWS EC2 Instance

Docker container

Python webserver

Prototype Review

nau.zeek.org

Challenges & Resolutions

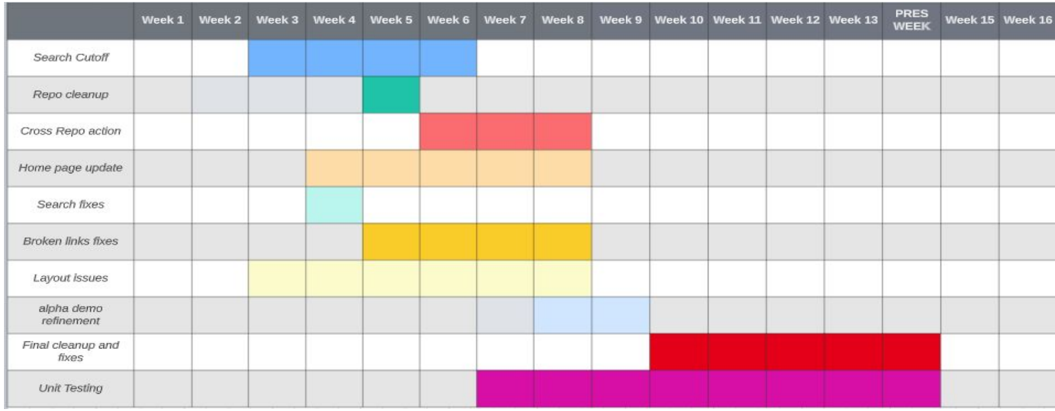
- Integrity of data sources
- Parsing and scraping of package data could return faulty results also causing the front end injection to be inaccurate
- Search engine could produce inaccurate results to users if original sources are also inaccurate
- Extensive and meaningful testing ensuring package data is injected properly and within an ideal time window will mitigate the potential risks associated with this project

Testing plan

- Unit Testing
- Integration Testing
- Scheduling time with Zeek Developers to perform usability testing across the whole site
  - Ensure information is easy to find
  - Ensure results are as expected from consumer audience

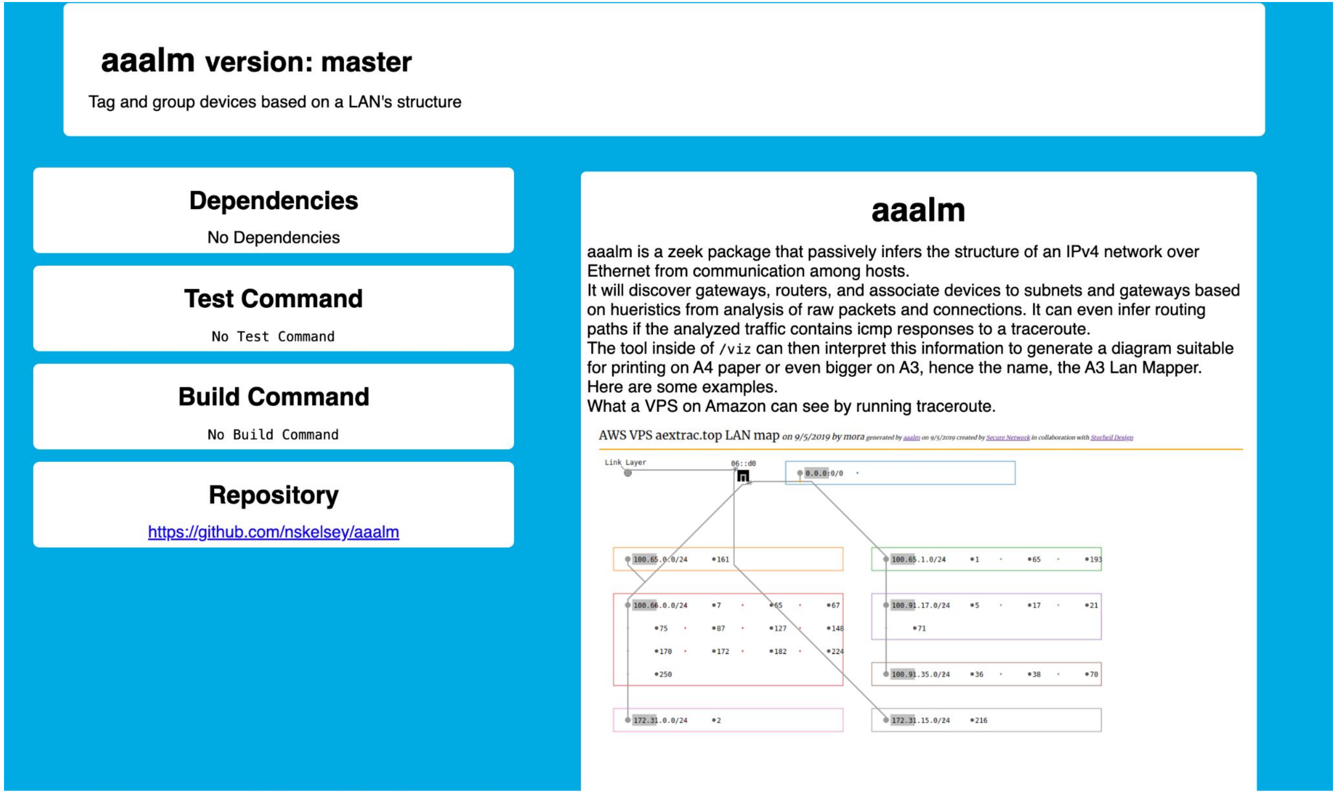
Upcoming Schedule

- Testing
- Fixing bugs and issues
- Refinement and collaboration with our client



	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	PRES WEEK	Week 15	Week 16
Search Cutoff			■	■	■	■										
Repo cleanup					■											
Cross Repo action						■	■	■								
Home page update				■	■	■	■	■								
Search fixes				■												
Broken links fixes					■	■	■	■								
Layout issues			■	■	■	■	■	■								
alpha demo refinement								■	■							
Final cleanup and fixes										■	■	■	■	■		
Unit Testing							■	■	■	■	■	■	■	■		

Wrapping Things Up



aaalm version: master

Tag and group devices based on a LAN's structure

Dependencies

No Dependencies

Test Command

`No Test Command`

Build Command

`No Build Command`

Repository

https://github.com/nskelsey/aaalm

aaalm

aaalm is a zeek package that passively infers the structure of an IPv4 network over Ethernet from communication among hosts.
It will discover gateways, routers, and associate devices to subnets and gateways based on hueristics from analysis of raw packets and connections. It can even infer routing paths if the analyzed traffic contains icmp responses to a traceroute.
The tool inside of `/viz` can then interpret this information to generate a diagram suitable for printing on A4 paper or even bigger on A3, hence the name, the A3 Lan Mapper.
Here are some examples.
What a VPS on Amazon can see by running traceroute.

AWS VPS aextrac.top LAN map *on 9/5/2019 by mora generated by aaalm on 9/5/2019 created by Secure Network in collaboration with Storheil Design*