Changing Network Detection
Using Bro and Distributed Computing Concepts

Mike Reeves @TOoSmOotH
“Who are you and why are you talking to me?”

- 16 years in InfoSec, 19 years total IT
- Work at FireEye
- Huge Bro fan
- Lots of experience in large sensor deployments
- Heavily into RC stuff.. FPV, Autonomous flight etc
- Security Onion contributor - Onion Salt
Story Time
Pyramid of Pain

*TM David Bianco
They know you better than you know yourself

• You will always have critical data at the edge

• They know the typical value prop of network detection/prevention is to centralize

• So you will get hit where you are the weakest
Network Detection is Awesome

- Packets don’t lie for the most part
- You can snag malware pre-detonation
- Quick way to get some sort of detection to hosts on your network with minimal disruption
- Host based stuff only works on things you have it installed on
Big Trouble in Little China

• Asynchronous Routing
• Encryption
• The “bad guys” are smart and will avoid you
• MPLS
• WAN Optimization
• Network Detection and the evolution of networking are in direct conflict
A”stink”ronous Routing

• Thanks BGP!

• The reality is you are going to hit this no matter what you do

• You should be doing asynchronous routing to improve network performance - just sucks for people trying to do detection
Encryption

• You are screwed for the most part
• Use SSL termination devices and put your sensors behind those
• There are commercial MITM products you can get some stuff out of
• Still screwed though
MPLS

• Network teams want to use MPLS the right way

• There is a lot of “hub and spoke” configurations still to centralize traffic for detection needs

• This ends up costing more money
WAN Optimization

• Cool concept and can save money

• Jacks up your detection capabilities since it only sends part of the traffic once its been accelerated

• Requires a sensor on the unoptimized side
I have told you everything that sucks. What should we do?

Time to flip detection on its head!
No more whiz bangery

• Choke points and the evolution of networking are in direct conflict

• If indicators are good enough for your sensors its good enough for everything

• Flexibility is the most important part of an effective detection strategy

• There is no “shiny and chrome” fix
How do we fix this?

• Go where the users are.. That is what the bad guys do

• Networks are distributed - so should your detection

• This means lots of Bro devices in lots of places

• Let’s steal some concepts from distributed computing
Introducing the double decker couch

• We use sensors for their resources.. like worker nodes in a HPC

• Workers can be rebuilt within minutes

• Should be able to run on whatever hardware is around

• The entire grid should be managed as a single device
How do we do this? By making our Bro sensors dumb!

- Pull off as much as possible
- No more atomic indicators on sensors
- Sensors are there to provide data to the backend
- Use low power devices
Minions work for their Masters

- Minions check in on a predetermined timeframe to ensure all things are like they are supposed to be.
- This allows us to have a single config for thousands of devices.
Demo Time
That’s neat.. But how do I detect stuff

- Break things into a service based architecture
- This makes scaling these services possible
- Forces data to be centralized instead of devices
- Puts the horsepower to detect lots of evil where its easy to scale vs sensors have finite resources
- Still need “deep packet inspection” to run on the sensors
Pub-sub to the rescue

- Ship bro logs from the sensors into some sort of Pub-Sub architecture. ex. REDIS, RabbitMQ
- Make subscribers process the log files looking for your indicators
- Expensive rules mean more work not lost packets
You mean I can use logs too?

- Atomic indicators work great on all kinds of different logs
- IP Addresses: bro_conn, firewall logs, proxy logs, webserver logs, host logs
- URI/URL: bro_http, proxy logs, web server logs
- Domains: bro_dns, dns logs, proxy logs, host logs
ESM
Enterprise Security Monitoring

- David Bianco BSides Augusta 2013
- https://www.youtube.com/watch?v=gA65N-RSWQ0
- @DavidJBBianco
What did we improve?

- MPLS can be used as intended. No more hub and spoke
- Asynchronous routing is no longer as much of an issue since we are closer to the users
- Ability to get traffic before it is optimized
- Gives you more eyes in more places to detect lateral movement
Questions?