Restricting device's network access using MUD files and automatic MUD file creation using OpenWrt®

Another approach trying to solve the missing „S“ in „IoT“

Disclaimer

This talk is about primarily IoT in the home - not about industrial IoT.

The technologies introduced might be – used with caution and reasonable care – useful for industrial IoT as well.

Still the same challenge with home IoT…

- IoT devices (especially in home networks) and their management software still have a „life on their own“:
  - penetrating firewalls
  - establishing random communication with weird sources in strange places in the world
  - randomly go crazy and get evil…

(Most) people don’t want that, right?

Some while ago…

… wise people came up with a simple approach:

- define the expected behaviour of an IoT device:
  - all IP connections a device is supposed to be established are described in a well-structured manner
  - a commonly used, easy to process text-based file format is chosen
  - manufacturers are encouraged to provide such files for the devices they sell

… and they called it…

M.U.D

MUD - IEEE RFC 8520

No, seriously - a great idea and approach!

- M.U.D - Manufacturer Usage Description - a simple JSON file describing a devices expected behaviour
- Easy processing
- Can be „hand-written“
- IEEE Standard (RFC 8520)

BUT…

MUD - IEEE RFC 8520

There’s still the „Chicken & Egg“ problem…

MUD - IEEE RFC 8520

What is the incentive for the device makers? Let’s face the facts:

- Government Authorities: No regulatory requirements to provide M.U.D files
- ISPs & Carrier’s: No clue what to do with IoT devices at home user’s places
- No unified community approaches to provide M.U.D files
- ....

Maybe another approach might help?

OpenWrt® … why?

- Defacto—industry standard for CPE’s
- Used as the default SDK by all major CPE chipset makers, although they call it „BDK“, „QSDK“…
- Estimated device roll-out per year: 200 - 300 Million devices
- Foundation for industry WiFi AP & CPE initiatives like prpl & TIP

Maybe there’s something in OpenWrt that might help?

Introducing unet-acl

unet-acl build to perform the following tasks:

- do client detection via
  - Automatic via DHCP snooping
  - Static configuration
- client MAC/IP tracking and enforcement an unregistered client’s traffic is discarded
- enforcement of per MAC bandwidth limit
- full traffic accounting
- (per day) traffic limits

Introducing unet-acl

its using firewall marks in conjunction with firewall rules to implement:

- Captive Portals: DNAT to local HTTP for un-authenticated traffic
- Parental Controls
- …

Introducing unet-acl

An unet-acl tracked interface has a set of rules/classes attached, that can contain directives to:

- Rewrite the egress (outgoing) interface
- Rewrite destination MAC
- Add FW mark
- Add/remove vlan
- …

Introducing unet-acl

allows automatic mapping of client -> rule via

- *any*
- Protocol/port
- Destination IP (or DNS snooped FQDN)
- (Any combination of the above)

Wait…

… this kind of sounds a bit like M.U.D, right?

unet-acl & M.U.D

Implemenation of M.U.D support is already in the works

- M.U.D files can be read by unet-acl and applied to devices, enforcing the pre-defined traffic patterns
- Devices are being monitored and activities out of the boundaries of the M.U.D definition can be used to create notifications (i.e. for uCentral)
- Traffic of devices that are not registered is discarded

unet-acl & M.U.D

Well, that's close - but no cigar, yet….

unet-acl M.U.D auto-learning

unet-acl implements an M.U.D auto-learning mode:

- allows adding a client with a blank MUD file – the service will monitor the client for a period of time and generate a MUD file based on observed behaviour
- MUD file then needs a manual review/verification
- automatic generation of client specific PCAP files for more detailed analysis

Is this the ultimate cure?

NO! …but

- Implements an industry standard approach to use M.U.D files for CPEs and Home Gateways - thanks to OpenWrt®
- M.U.D auto-learning mode is a starting point for researchers and communities to provide manufacturer-independent M.U.D information
- adoption can create community based data bases of devices – ok, let's dream a little bit here ;)

Thank you!

Questions…