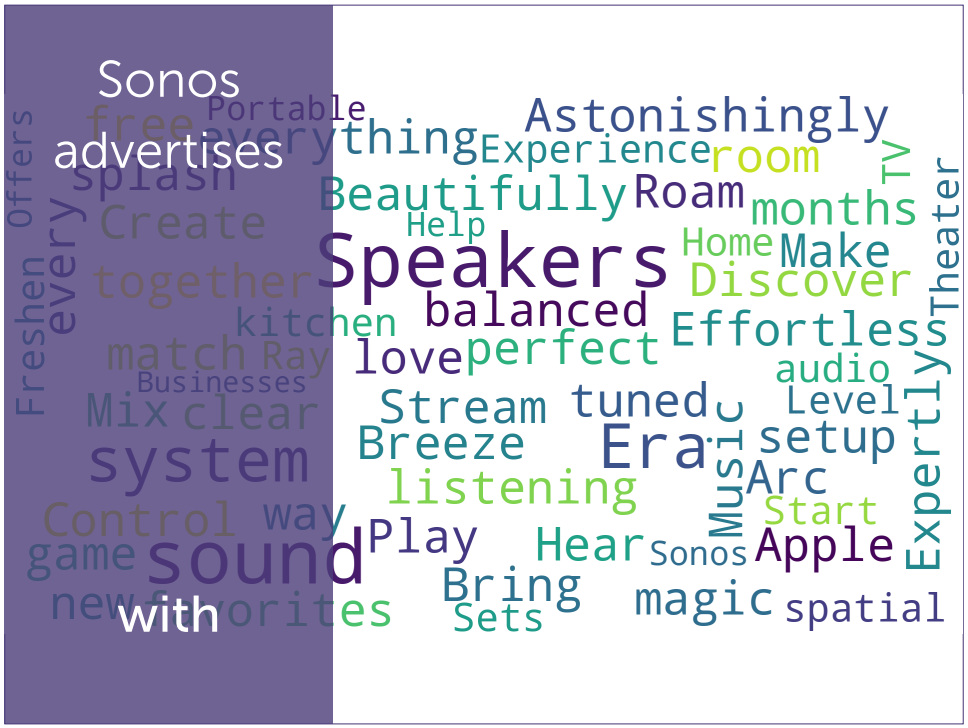
Universität Basel

Sonos Project

Ugur Turhal – ugur.turhal@unibas.ch

15th May, 2023



Introduction

Figure 1: Illustration of my Project

Structure

- Scan the network with `nmap`
- Select in the network the Sonos speakers as targets
- Provide the Sonos `ID` in the network
- Command them with `SOAP`

General - Commands

Function	Action
`--help`	list the help function
`--target`	Select the Sonos boxes as target
`--ltargets`	show all targets
`--refresh`	scan all devices in network
`--sonos`	show Sonos devices
`--commands`	show Sonos commands
`--exit`	stop the program
`--csv`	get a csv of the current list of devices in the network and a csv with the open `ports/mac/ip`

Table 1: Commands

`--commands`

Sonos Command	Action
`play`	play a song
`next`	skip to the next song
`previous`	play the previous song
`pause`	pause the song
`queue` + `LINK`	queue a song from Spotify
`mute` + `0 | 1`	mute the box(es)
`volume` + `args[0:100]`	set the volume of the box(es)

Table 2: Commands

Sonos control

Control via Soap

1. To control the sonos speaker `SOAP`[1] protocol is used.
2. Used `SOAP call` (`an HTTP request`), with some special headers and some XML formatted body.
3. Each request is a `POST request` to a control endpoint in my case it is (for play, pause, next, previous, queue): `POST /MediaRenderer/AVTransport/Control HTTP/1.1`.
4. Important: Each `request` is made to the port `1400`

Figure 2: SOAP, is not just used for washing hands.[2]

[1] Simple Object Access Protocol

[2] Still frame: https://www.prevention.com/health/g31965281/best-hand-soaps/

Example

SOAP - Structure

```
POST /MediaRenderer/AVTransport/Control HTTP/1.1
CONNECTION: close
ACCEPT-ENCODING: gzip
HOST: {ip}:1400
USER-AGENT: Linux UPnP/1.0 Sonos/62.1-86220 (WDCR:Microsoft
    Windows NT 10.0.19042)
CONTENT-LENGTH: 252
CONTENT-TYPE: text/xml; charset="utf-8"
X-SONOS-TARGET-UDN: uuid:{uuid}
SOAPACTION: "urn:schemas-upnp-org:service:AVTransport:1#Pause"
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="..." s:encodingStyle="...">
  <s:Body>
    {ActionBodyHere}
  </s:Body>
</s:Envelope>
```


Command line interface - 1

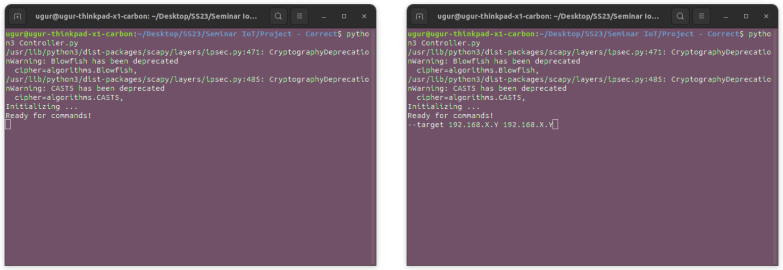


Figure 3: Left: Start of the Programm Right: Targeting the boxes

Command line interface - 2

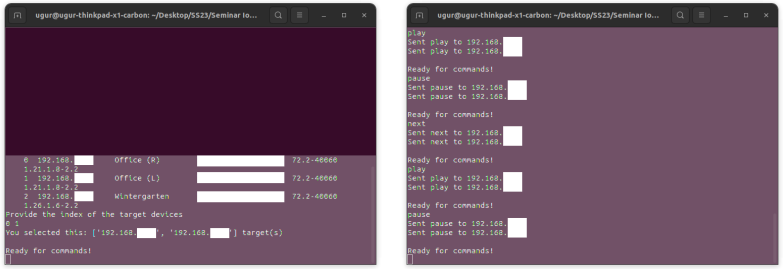


Figure 4: Left: Index of the Sonos boxes. Right: Commanding

Traffic analysis - setup

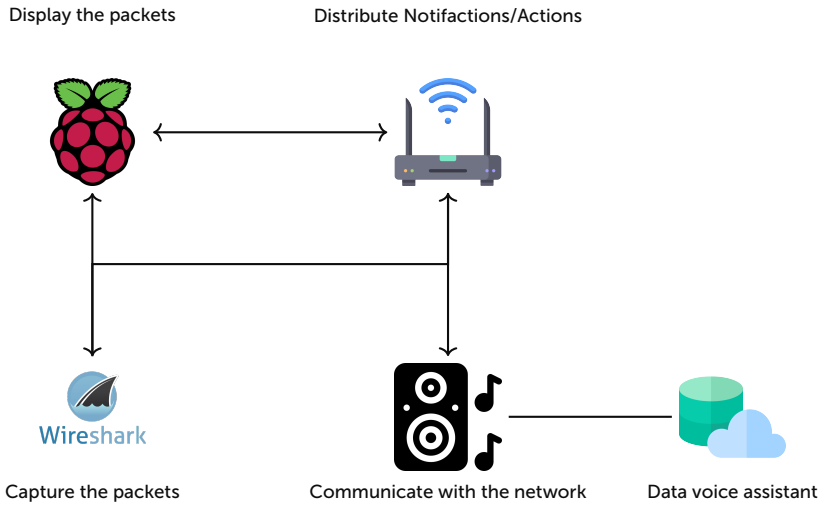


Figure 5: Setup for capturing packets

120 hours - Result

Figure 6: **Left:** Traffic analysis for 24 hours, Right: Traffic analysis for 120 hours

Conclusion

Observation

- ☐ Sonos - VA: has a lot more traffic, but two boxes, communication is over SSDP. The Keyword: Hey Sonos uses WOL. The wake-on-LAN protocol enables the device to **wake up** from a **sleeping** or **powered off** state.
- ☐ Google - VA: is installed on the third Sonos device, from the same manufacturer same model. Uses MDNS for communication. The device **does not use WOL**, which means that Google, in theory, could do unrecognized recordings. Even if the keyword is not used.

Conclusion

- ☑ Controlling Sonos boxes, is functioning flawlessly.
- ☑ Traffic analysis shows that only music control commands are recorded and sent. (Sonos VA)
- ☑ Thousands of packets are very much!
- ☑ I can send notifications from my laptop to every Sonos Box. $\Rightarrow$ Open for malicious attacks.