

Amazon Halo Network



INST346 | Technologies, Infrastructure and Architecture

Group 24

Tristan Clark | Cybersecurity Engineer & Data Analyst

Jomer Paulino | UX Researcher & Network Engineer

Asad Raheem | UX Designer & Scrum Leader

Brandon Sapp | Solutions Architect & Software Developer

The Problem and The Persona

Persona

People with physical fitness goals, people who regularly exercise, people who wish to track their fitness

Problem

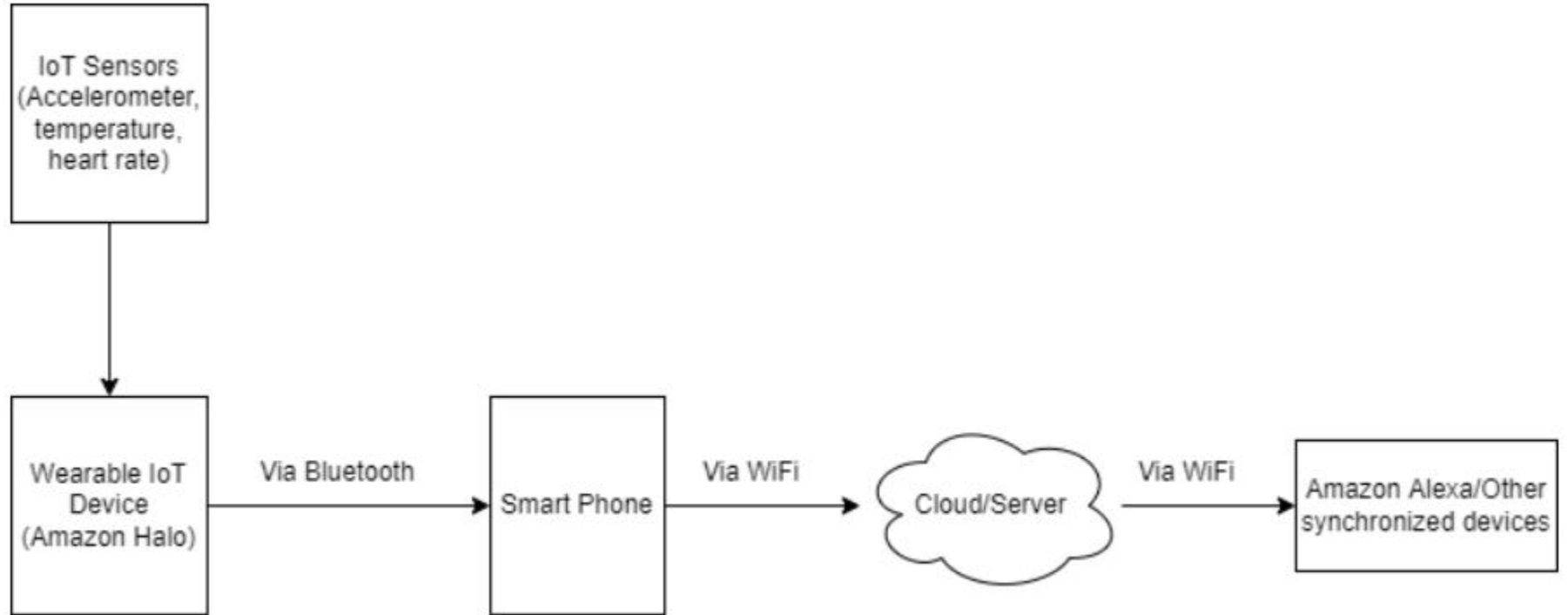
Hard to maintain your fitness or achieve goals if you can't track your progress

Solution

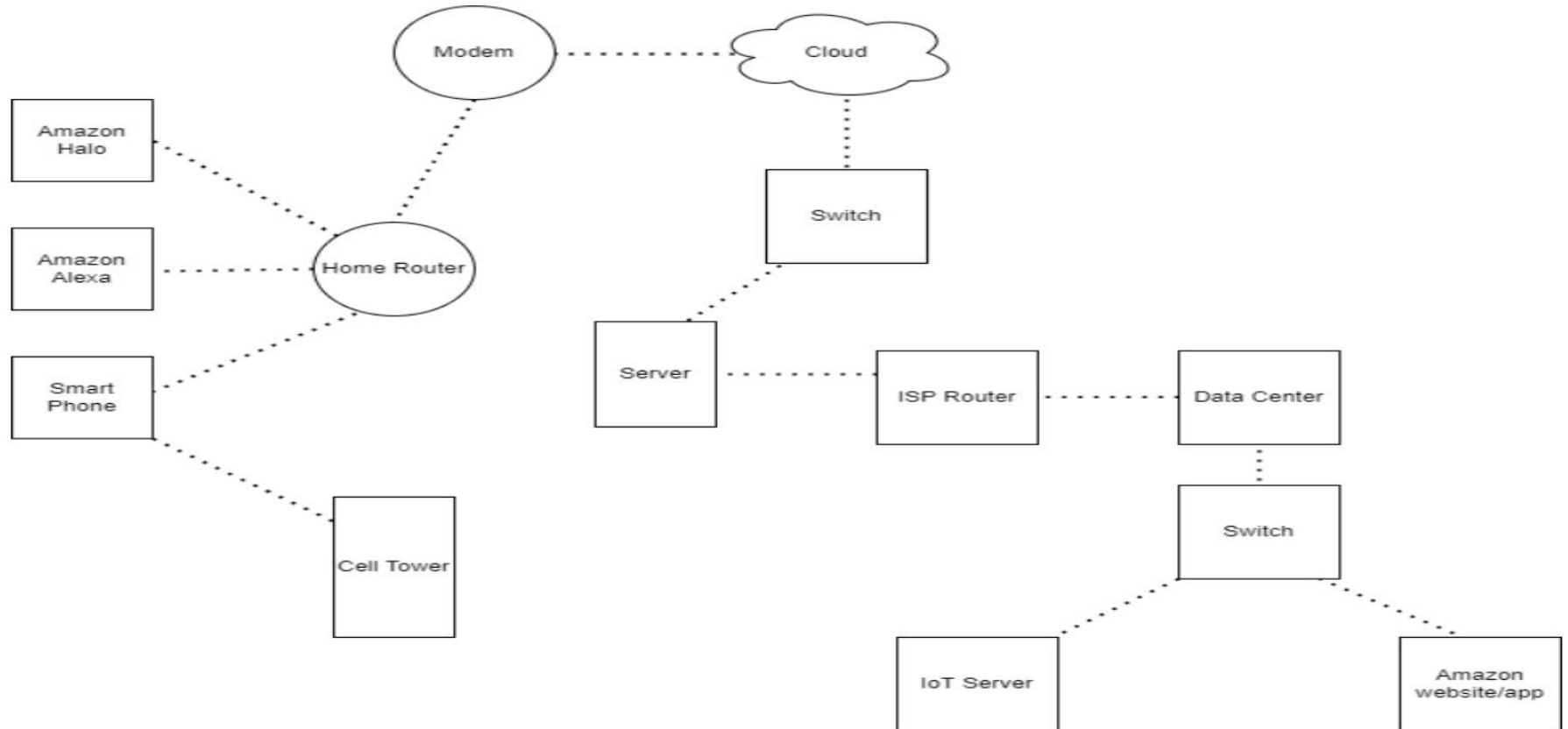
Automatically track daily fitness and activity via a wearable IOT device



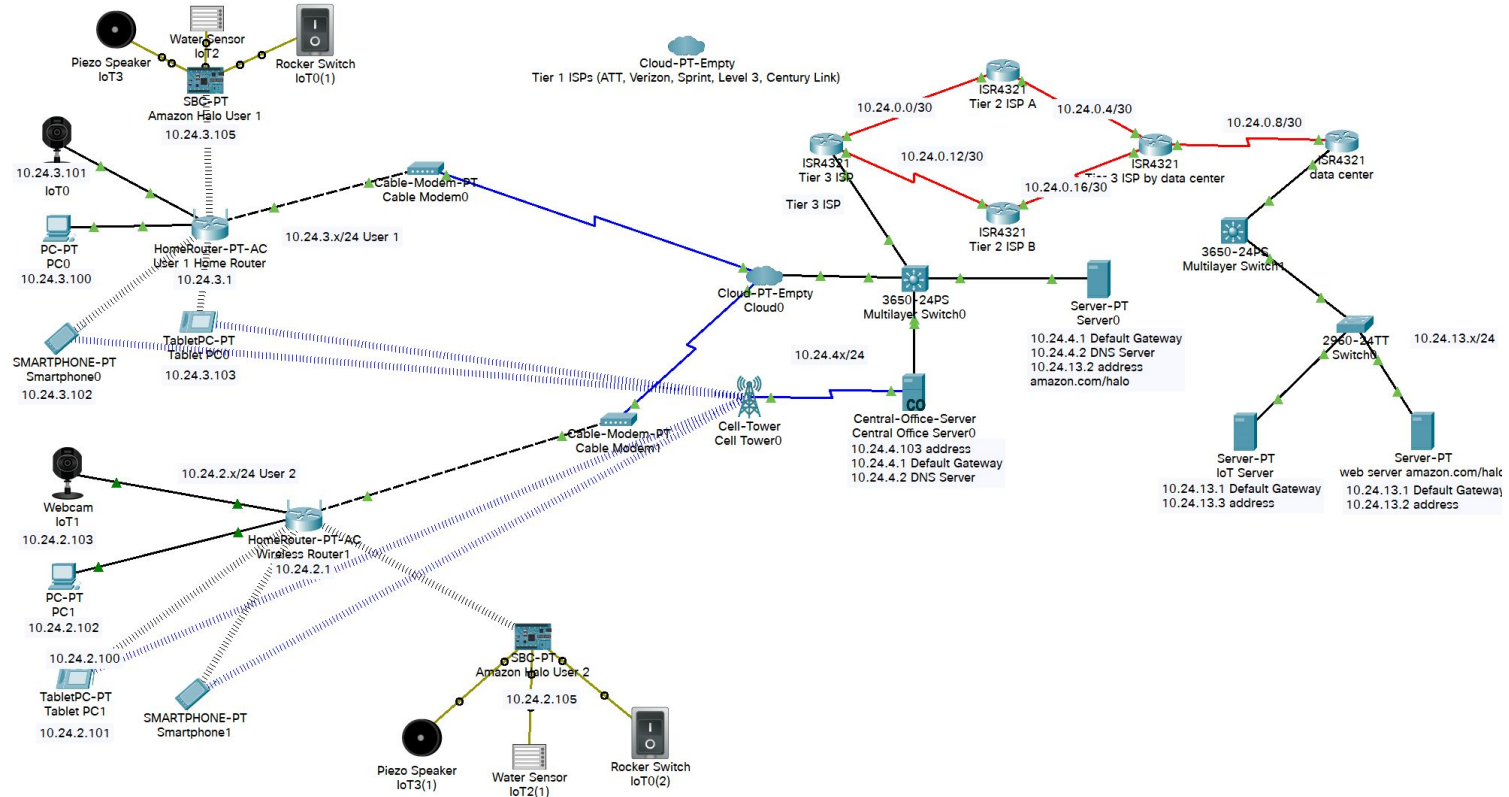
High Level Block Diagram

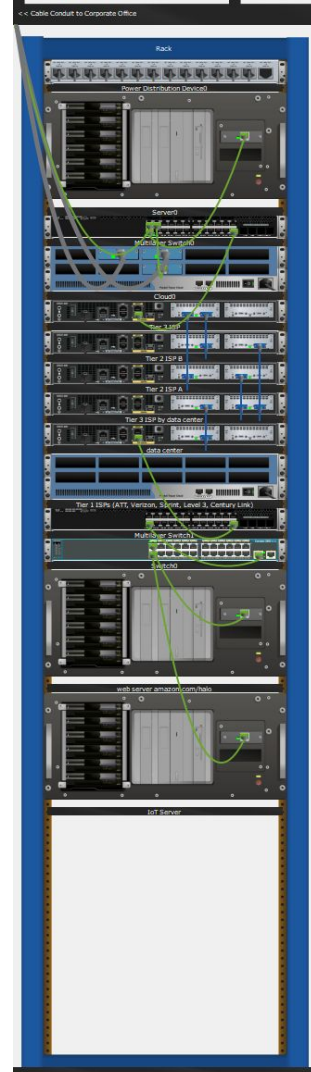
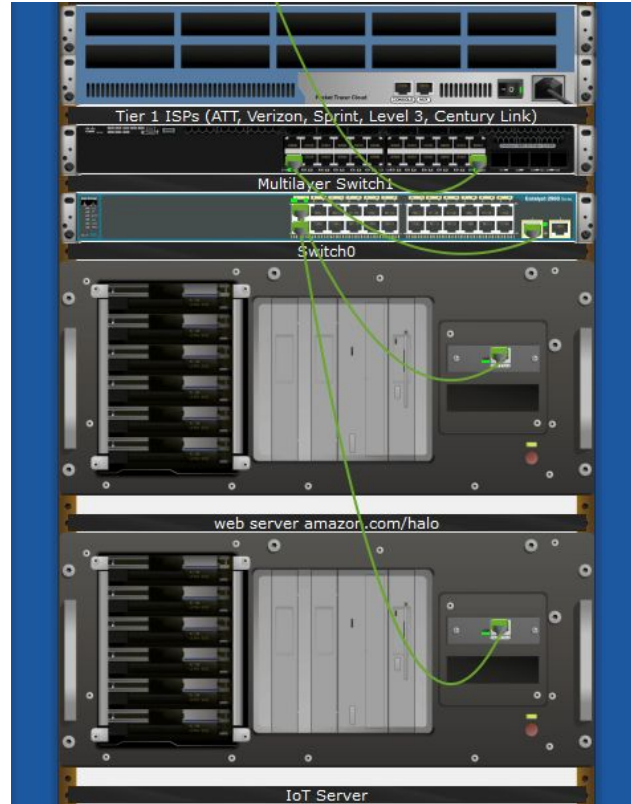


Network Diagram



Logical Topology





4 Vs of Big Data Applied IoT Data

- Volume

- Source - 2Gb
- Cloud - 50Tb

- Velocity

- Source - 1Mb/minute
- Cloud - 10Gb/minute

- Variety

- Source - Structured data all from one source
- Cloud - All structured data of combined Halo devices

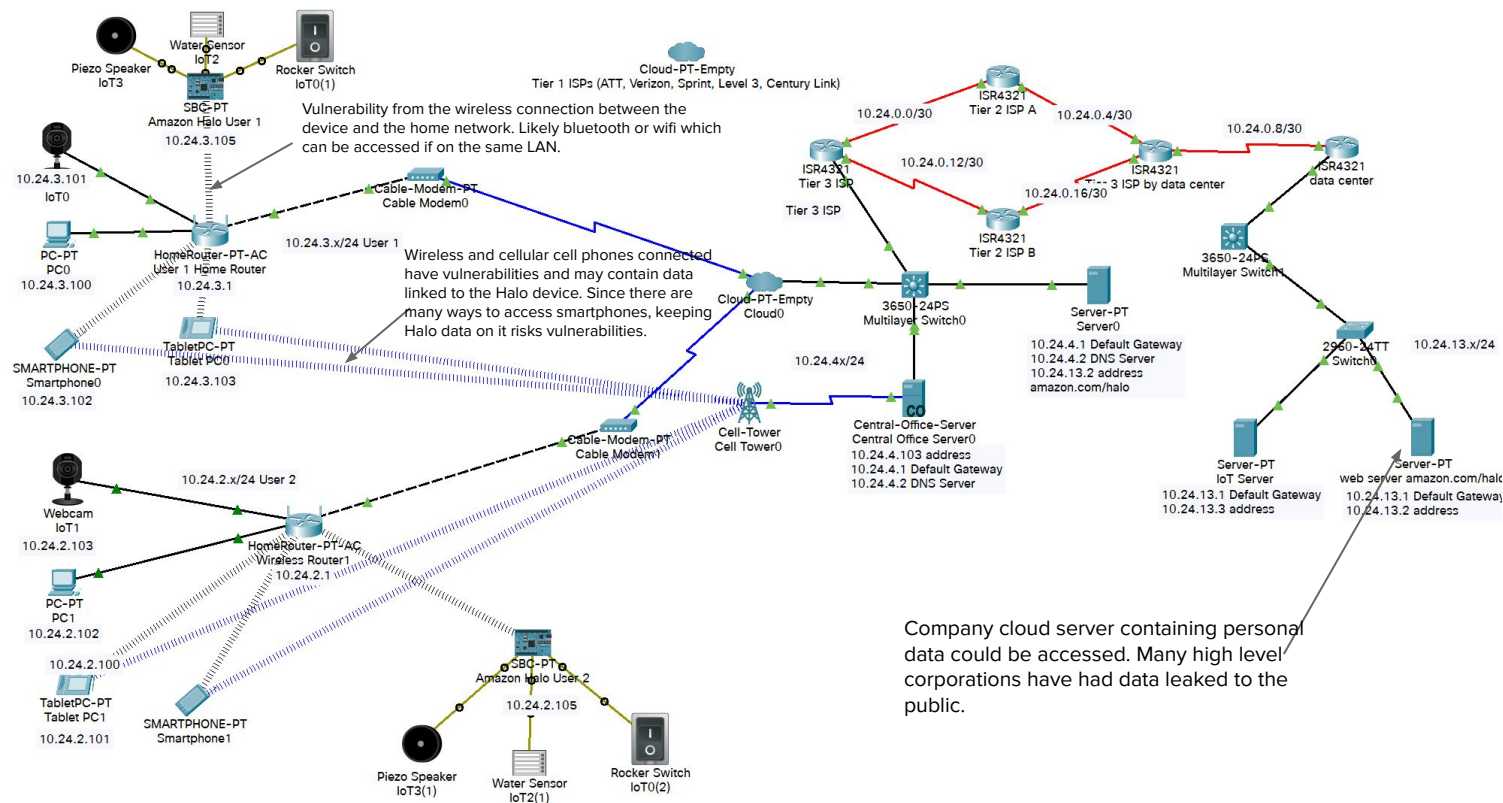
- Veracity

- Source - Given the lack of variety, the certainty of the data will be very high
- Cloud - Will have similar certainty of data as source due to collecting same data from multiple sources

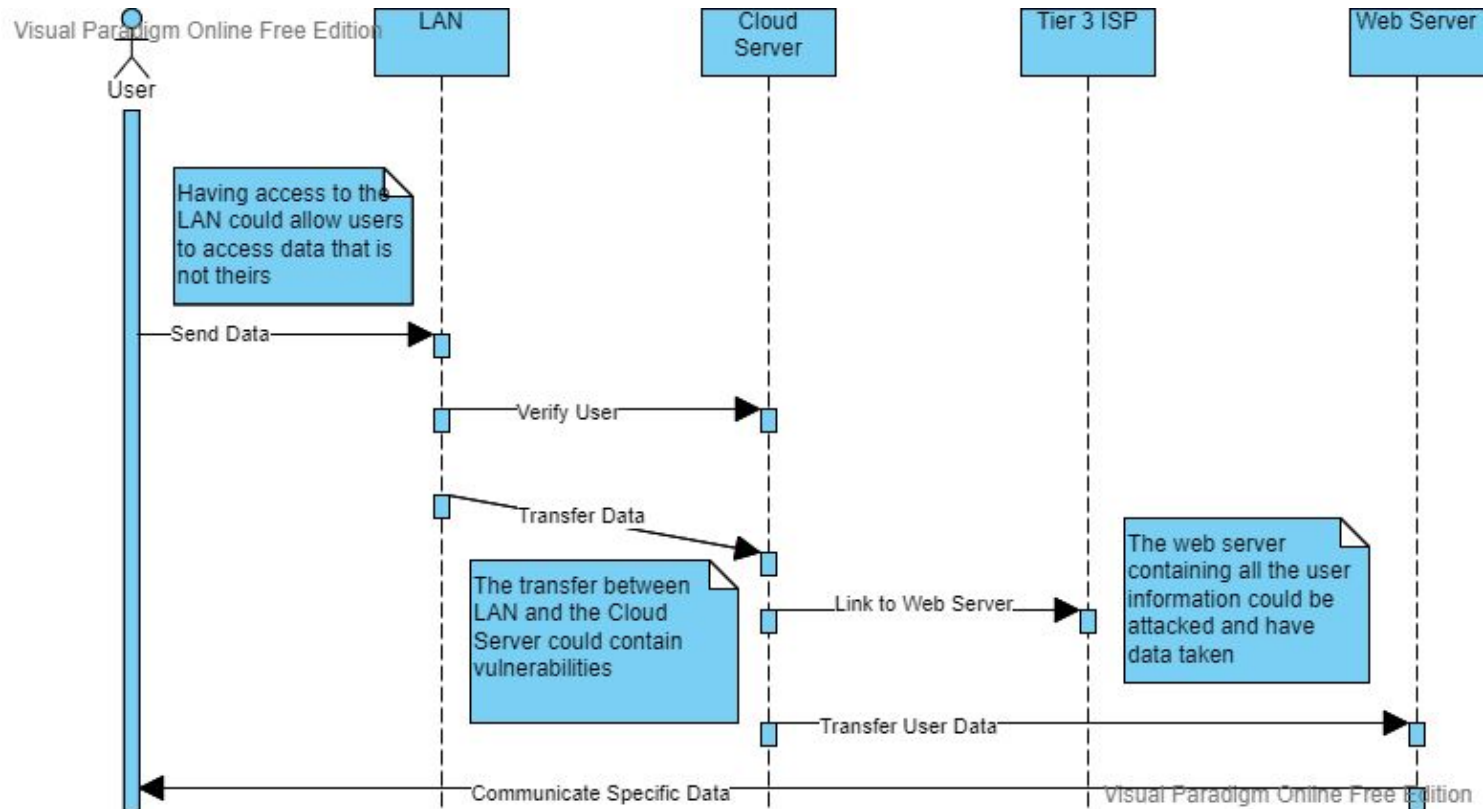
Scale this network

- One method to increase scalability within the network would be to add cellular to the device
- Giving the device the ability to use a cellular network would eliminate the need for a LAN
- Integrating the processing of data to the device rather than the cloud would allow any amount of users to connect
- Increasing the amount of web servers available to accommodate future devices
- The network is already quite scalable and could be upgraded seamlessly by adding more processors and devices

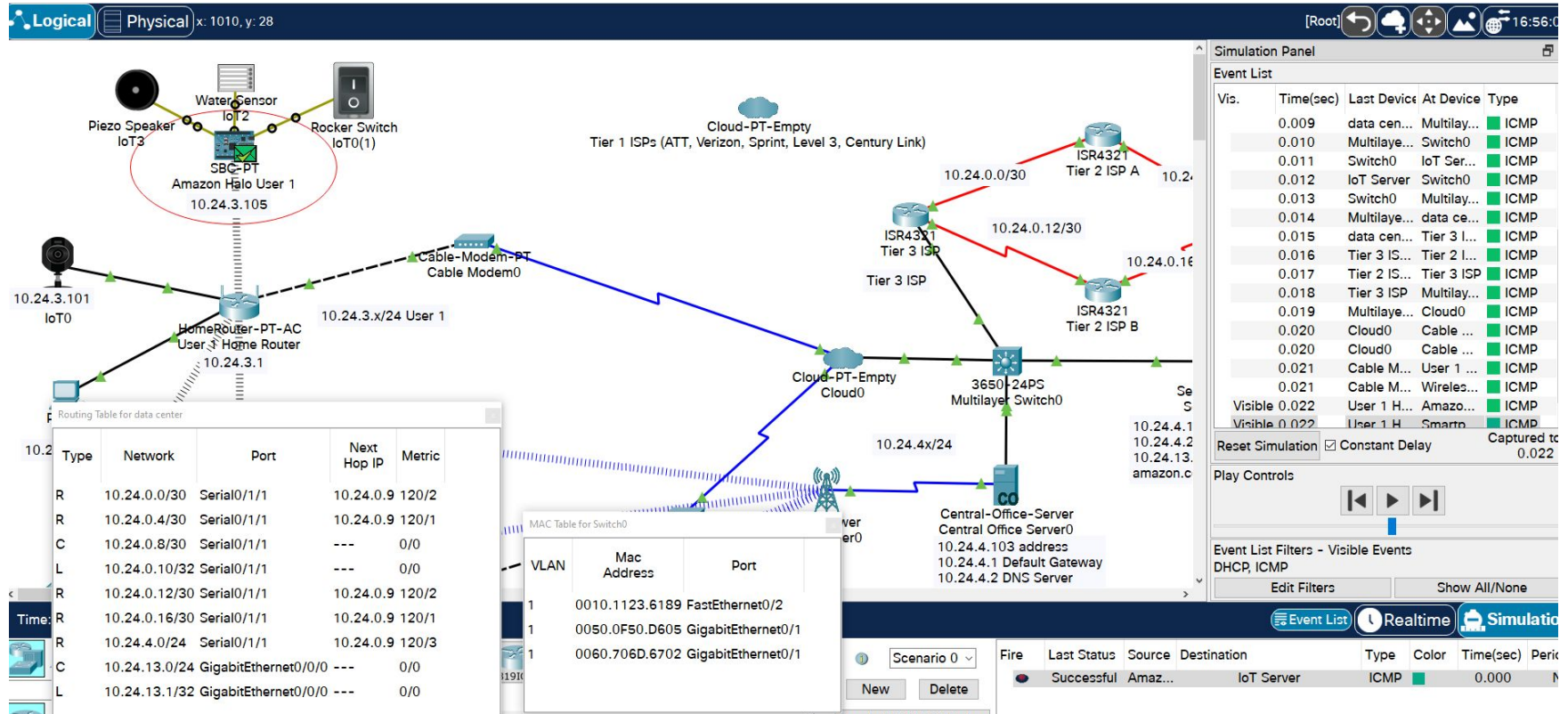
Annotated PT Logical Topology Showing Vulnerabilities



Sequence Diagram of IoT Interaction



Ping 1 (User to IoT Server)



HTTPS Web Page request and response

The screenshot displays the Cisco Packet Tracer interface. On the left, a network diagram shows a Home Router (10.24.3.1) connected to various IoT devices (Piezo Speaker, Water Sensor, Rocker Switch) and a PC (10.24.3.100). The PC is connected to the Home Router, which is connected to the Internet (10.24.3.x/24 User 1). The Home Router is also connected to a Tablet PC (10.24.3.103) and a Smartphone (10.24.3.102). The Home Router is connected to the Internet (10.24.3.x/24 User 1). The Home Router is connected to the Internet (10.24.3.x/24 User 1). The Home Router is connected to the Internet (10.24.3.x/24 User 1).

In the center, a web browser window titled "Amazon Halo User 1" shows the URL `http://10.24.13.3` and the page content: "Cisco Packet Tracer", "Welcome to Amazon Halo. Opening doors to new opportunities. Brought to you by Group 24!", and "Quick Links: [A small page](#), [Copyrights](#), [Image page](#), [Image](#)".

On the right, the "Simulation Panel" shows the "Event List" with the following data:

Vis.	Time(sec)	Last Device	At Device	Type
	0.041	Switch0	Multilay...	HTTP
	0.042	Multilay...	data ce...	HTTP
	0.043	data cen...	Tier 3 l...	HTTP
	0.044	Tier 3 IS...	Tier 2 l...	HTTP
	0.045	Tier 2 IS...	Tier 3 ISP	HTTP
	0.046	Tier 3 ISP	Multilay...	HTTP
	0.047	Multilay...	Cloud0	HTTP
	0.048	Cloud0	Cable ...	HTTP
	0.048	Cloud0	Cable ...	HTTP
	0.049	Cable M...	Wireles...	HTTP
	0.049	Cable M...	User 1 ...	HTTP
Visible	0.050	User 1 H...	Tablet ...	HTTP
Visible	0.050	User 1 H...	Amazo...	HTTP
Visible	0.050	User 1 H...	Smartp...	HTTP

The "Simulation Panel" also includes "Play Controls" (Reset Simulation, Constant Delay, Captured to: 0.050 s) and "Event List Filters - Visible Events" (DHCPv6, DNS, EIGRPv6, HSRPv6, HTTP, HTTPS, ICMPv6, IoT, IoT TCP, NDP, OSPFv6, PTP, RIPng, STP). The bottom status bar shows "Time: 00:03:16.592" and "PLAY CONTROLS".

IoT Registration request and response

The screenshot displays a network simulation environment with two primary windows open:

- Amazon Halo User 1 Configuration Window:**
 - GLOBAL Settings:** Algorithm Settings, Files, INTERFACE (FastEthernet0, Wireless3, Bluetooth).
 - Gateway/DNS Server:**
 - ☐ DHCP
 - ☒ Static
 - Default Gateway: 10.24.3.1
 - DNS Server: 10.24.4.2
 - Gateway/DNS IPv6:**
 - ☐ Automatic
 - ☒ Static
 - Default Gateway: [empty]
 - DNS Server: [empty]
 - IoT Server:**
 - ☐ None
 - ☐ Home Gateway
 - ☒ Remote Server
 - Server Address: 10.24.13.3
 - User Name: admin
 - Password: admin
 - Refresh button

- IoT Server - Devices Window:**
- IoT Monitor:** Home | Conditions | Editor | Log Out
- IoT Server - Devices:**
 - IoT1 (PTT08108U30-) Webcam
 - Amazon Halo User 1 (PTT08109WLK-)** (highlighted with a red box)
 - Amazon Halo User 2 (PTT0810105W-) Webcam
 - IoT0 (PTT0810PYVL-) Webcam

The background shows a network topology diagram with various devices connected. A red circle highlights the 'Server-PT IoT Server' device in the topology, which is associated with the IP address 10.24.13.1 and the default gateway 10.24.13.2.

At the bottom, a status bar shows the time as 00:07:12 and a table of network events:

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	--	IoT0	web serv...	ICMP		0.000	N	0	(edit)	(delete)

API Code and Results

```
web server amazon.com/halo
Physical Config Services Desktop Programming Attributes
workout (Python) - main.py
Open New Delete Rename Import
Reload Copy Paste Undo Redo Find Replace Zoom: + -

main.py
1 # wgr contains a database with a large variety of exercises and
2 # dietary information for fitness purposes
3
4 # Import libraries
5 from realhttp import *
6 from time import *
7 import json
8 # Initialize URL to access exercises from the wger exercise database
9 url = "https://wger.de/api/v2/exercise"
10
11 # Define functions
12 def onHTTPDone(status, data):
13     #print("status: " + str(status))
14     #print("data: " + data)
15
16     # take raw json data and parse it
17     json_object = json.loads(data)
18     # Use json.dumps and indentation to make json format readable
19     print(json.dumps(json_object, indent = 1))
20
21 def main():
22     # http client (what PT server uses to make http request)
23     # create callback, and url request
24     http = RealHTTPClient()
25     http.onDone(onHTTPDone)
26     http.get(url)
27
28     # don't let it finish
29     while True:
30         sleep(3600)
31
32 # Execute
33 if __name__ == "__main__":
34     main()

"exercise_base": 41,
"status": "2",
"description": "<p>Place a barbell on the floor at your feet.</p><n><p>Bending at the waist, grip the
barbell with a shoulder with overhand grip.</p><n><p>With a slow controlled motion, roll the bar out so that
your back is straight.</p><n><p>Roll back up raising your hips and butt as you return to the starting
position.</p>",
"creation_date": "2015-07-27",
```

