proceedings

MDPI

Proceedings

Security Vulnerability Assessment of Google Home Connection with an Internet of Things Device †

Hilal Çepik [1], Ömer Aydın [2,*] and Gökhan Dalkılıç [1]

[1] Computer Engineering Department, Faculty of Engineering, Dokuz Eylul University, 35220 Konak, Turkey; hilal.cepik@ceng.deu.edu.tr (H.Ç); dalkilic@cs.deu.edu.tr (G.D.)
[2] Electrical and Electronics Engineering, Faculty of Engineering, Manisa Celal Bayar University, 45140 Manisa, Turkey
* Correspondence: omer.aydin@cbu.edu.tr; Tel.: +90-236-201-2164
† Presented at the 7th International Management Information Systems Conference, Online, 9–11 December 2020.

Abstract: With virtual assistants, both changes and serious conveniences are provided in human life. For this reason, the use of virtual assistants is increasing. The virtual assistant software has started to be produced as separate devices as well as working on phones, tablets, and computer systems. Google Home is one of these devices. Google Home can work integrated with smart home systems and various Internet of Things devices. The security of these systems is an important issue. As a result of attackers taking over these systems, very serious problems may occur. It is very important to take the necessary actions to detect these problems and to take the necessary measures to prevent possible attacks. The purpose of this study is to test whether an attack that attackers can make to these systems via network time protocol will be successful or not. Accordingly, it has been tried to attack the wireless connection established between Google Home and an Internet of Things device over the network time protocol. Attack results have been shared.

Keywords: Google Home; Internet of Things; wireless connection; security; privacy; security threats; security attacks; network time protocol; NTP; SNTP; IFTTT

Citation: Çepik, H.; Aydin, Ö.; Dalkiliç, G. Security Vulnerability Assessment of Google Home Connection with an Internet of Things Device. *Proceedings* **2021**, *74*, 1. https://doi.org/10.3390/proceedings2021074001

Published: 2 March 2021

1. Introduction

The Internet of Things (IoT) concept has been accepted as a stream of innovation. The use of sensors, which is very important in IoT technology, is increasing day by day. In this way, users can be provided with many conveniences in daily life. IoT technology is used in homes, industrial automation, and smart cities [1].

Using devices with IoT technology in homes makes homes smart. With these devices connected to the Internet, users can control and manage their homes and workplaces remotely. For example, it can adjust the temperature of the room, control security systems in the home or workplace, control lighting systems, monitor data from various sensors (heat, light, humidity, and smoke). With the increasing number of these devices, users' privacy and security concerns are increasing. For example, cameras and microphones built into surveillance equipment can be used by intruders to monitor domestic secret life [2].

One of the most popular devices used in smart homes and offices with IoT technology is a virtual personal assistant. An example of these assistants is Google Home. It is a smart speaker. Google Home is the embodiment of the Google Assistant. The user is only allowed to command the system with sound. The protection of the Voice User Interface is troublesome because it is difficult to authenticate the persons involved in the audio channel. Google Home is also one of the popular IoT devices targeted by attackers, because with Google Home, you can control smart devices and systems in an environment. An

attacker could access all other devices through this device and cause serious damage to the user.

One possible attack on Google Home is voice squatting attack (VSA). For example, a user sends a request to Google Home in the form of "Ok Google open Capital One". Attackers are starting an incorrect skill by enabling other requests similar to the request's pronunciation to be executed. For example, instead of "Capital One", "Capital Won" or "Caption One" can be understood. Today's speech recognition techniques are advanced but not yet perfect. Therefore, depending on the pronunciation of the speaking people and the noise factor in the environment, various weaknesses can be seen in the system [3].

One of the most popular technologies today is "if this then that" (IFTTT) technology. Together with Google Home, this technology makes life easier for users. The services used with this platform can be connected to each other and make it very easy to use. Much more can be done on applications such as Twitter, Dropbox, Evernote, Amazon Alexa, and Google Assistant on the IFTTT platform. For example, if you change your profile photo on Facebook, change your profile photo on Twitter.

Using Google Home and IFTTT together, the same operation can be done with smart devices in the environment. For example, turn on the lights when the door is opened or turn on the fans when the lights are opened.

There may be some issues with privacy and security issues on this platform. An attacker could manipulate a user's recipes by accessing a user's IFTTT account. This is a very serious problem. Since this platform can access all the smart physical devices in the smart home, the attacker who enters the platform can change the recipes of these devices or add new ones [4].

As a solution to the possible security problems, researchers have designed different solutions. The authors in [5] conducted an analysis. Based on this analysis, they designed a trigger-action platform. The name of this platform is Decoupled-IFTTT. On this platform, users do not have to give privileged access to their online services. As a result of the evaluations of the people who made this study, it was found that IFTTT had a small burden. A delay time of less than 15 ms was added to the recipe execution time, and the yield decreased by 2.5%.

IoT devices may need time values to communicate other devices securely. Therefore, there is a network time protocol or simple network time protocol for lightweight devices. With these protocols, devices reach a time server and get the time value from there. Changing the time value on the network or blocking it can cause security vulnerabilities. Some of the attackers use this method to disable or infiltrate the system. The purpose of this study is to test whether the attacker can affect the system with the network time protocol (NTP) attack. Moreover, this article explores how Google Home can securely communicate with an IoT device. Whether this communication is vulnerable to NTP attack has been tested. An IoT device has been used for this process. This device was later used to communicate with Google Home using IFTTT technology. Finally, by examining the traffic on the network, it was examined whether the time information obtained from NTP can be changed.

2. Materials

We used some materials to propose our study. In this section, these materials are given. A Node MCU, 5 V 2-channel relay module, 5 V adapter for Node MCU, jumper wires female to female, lamp, Android phone, and a laptop/personal computer will be investigated in the subtitles below.

2.1. Node MCU

It is a small electronic circuit that operates at low voltage (5 V) [6,7]. It has a Wi-Fi module, so it is preferred in making IoT projects. HTTP libraries can be used to communicate

with this device over the Internet. The reason why Node MCU is preferred over Arduino is that there is no Wi-Fi module in Arduino. The NodeMCU schema can be seen in Figure 1.

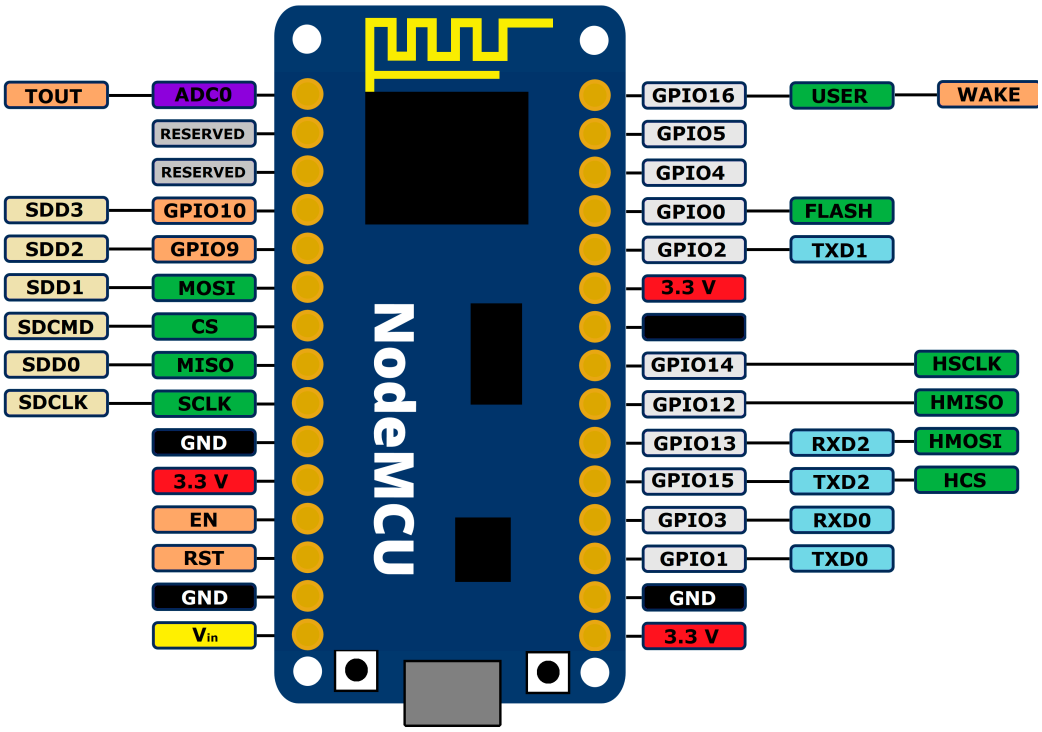

Figure 1. NodeMCU schema.

Its firmware is based on ESP8266 Wi-Fi SoC that provides Internet connectivity to the hardware or circuit along with it.

2.2. Relay Module

The relay is used for switching high-current devices using low current. It has a coil inside. When the coil of the relay is energized, the magnetic field is generated and pushed forward to allow the contacts to touch each other. When the voltage is cut off, the contacts are separated. If you want to operate a 220-volt device, the relay is given 5 volts. Then, the relay acts as a switch [8].

2.3. Google Home

Google Home is a home assistant equipped with a speaker, microphone, and camera. It answers voice search queries [9]. It works with the "ok google" command. It is integrated with Google Assistant and can manage smart home systems. Examples:

Automatically switching lights on and off (Smart sockets).
Setting the home temperature (Smart thermostat).
Weather information.
Playing music.
Set up a timer to automatically turn on and off electronic devices.

The circuit diagram of the prepared lamp is shown in Figure 2.

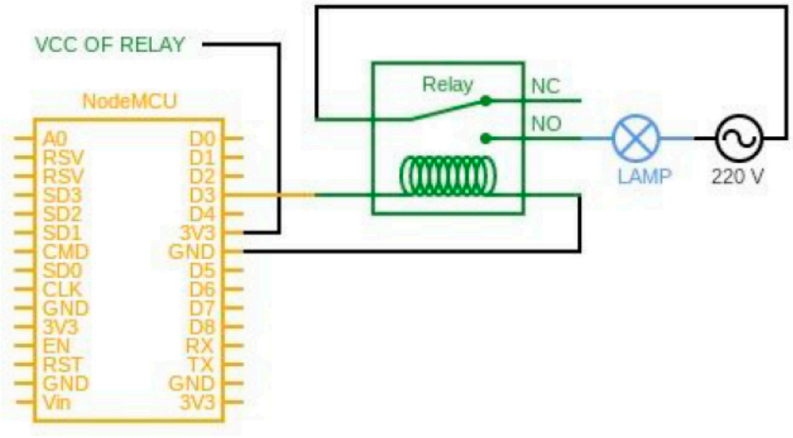

Figure 2. Circuit diagram of lamp.

The connection between the node module and the relay module pins in this circuit is shown in Table 1.

Table 1. Connection pins.

Relay Module	NodeMCU
GND	G
In1	D3
V_{CC}	3 V

2.4. Blynk App

Blynk application is an environment in which we can prepare applications that we can control the IoT devices we designed [10]. It supports many open source and popular equipment, not just Arduino. With this application, we can create an application for our IoT device without needing much hardware and software knowledge. In order to use, you must first become a member. This can be done by downloading the application from the store to a smartphone or from https://blynk.io/ (accessed on 28 August 2020) website. After the registration process is completed, a token is sent by Blynk via e-mail. By adding this authentication code to the code written for the IoT device, connection with the Blynk application is provided. Its usage is shown in Figure 3.

```
char auth[] = "**********************************";
Blynk.begin(auth, ssid, pass);
```

Figure 3. Internet of Things (IoT) Blynk authentiction code.

Once the connection is established, a new project can be created by opening the Blynk application. The application interface is shown in Figure 4. The D3 pin is selected in the Blynk application because the D3 pin of the NodeMCU is connected to the relay. The button in the application switches the lamp on and off. When the button is clicked, the D3 pin of NodeMCU is updated to 1 or 0.

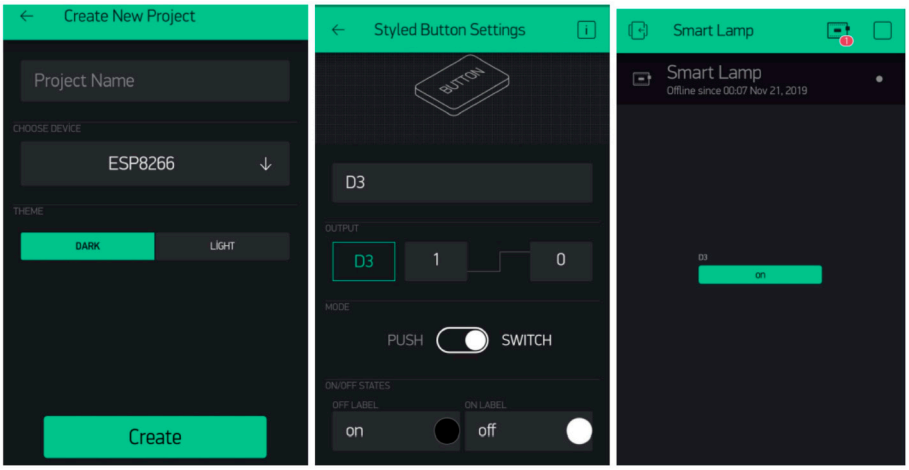

Figure 4. Blynk application interface.

2.5. IFTTT

IFTTT simply means If This Then That [11]. It is used after becoming a member. It connects the services we have used. It automates and simplifies many tasks. For example, when my Twitter profile changes my photo, it changes my Facebook profile photo, too. In this project, according to the voice command sent to Google Home, the Blynk application will update the status of the D3 pin in NodeMCU. Google Assistant and Webhooks services in IFTTT were used for these operations.

Firstly, three discourses are determined for the Google Home voice command. Then, the discourse to be said as the answer is determined. Figure 5 shows the trigger for turning on the light.

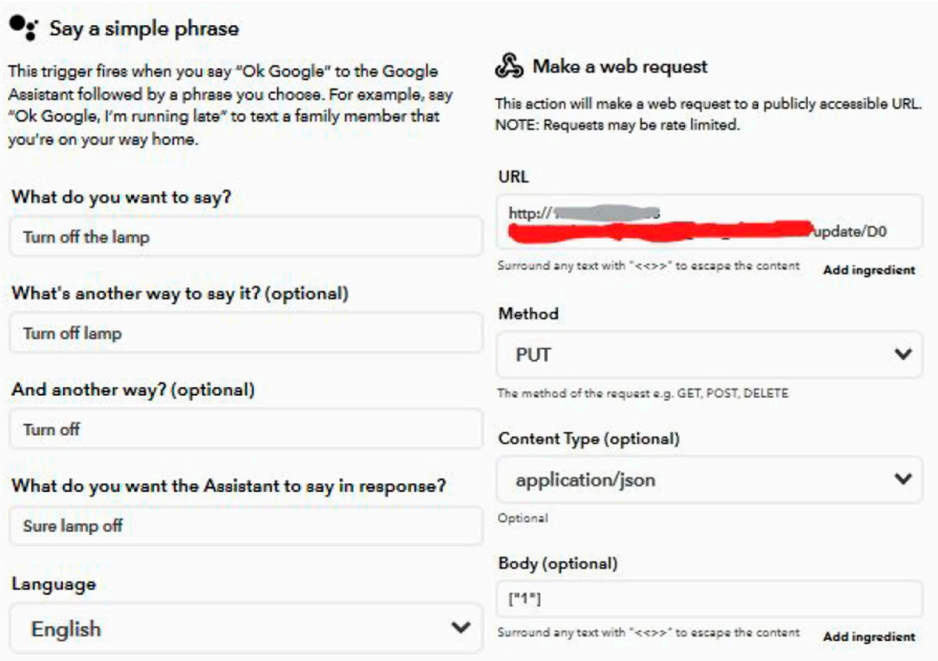

Figure 5. If This Then That (IFTTT) trigger turn on.

Figure 6 shows the trigger for turning off the light. When a command comes to Google, a trigger runs. Then, it runs a web request. The form of the web request is as follows:

http://IP/token/update/PIN (accessed on 28 August 2020)
IP: Blynk cloud login IP address

Token: Blynk Authentication Code of your project

IP depends where you are in the world. So, it can be learned by running the "ping blynk-cloud.com" command on the command line.

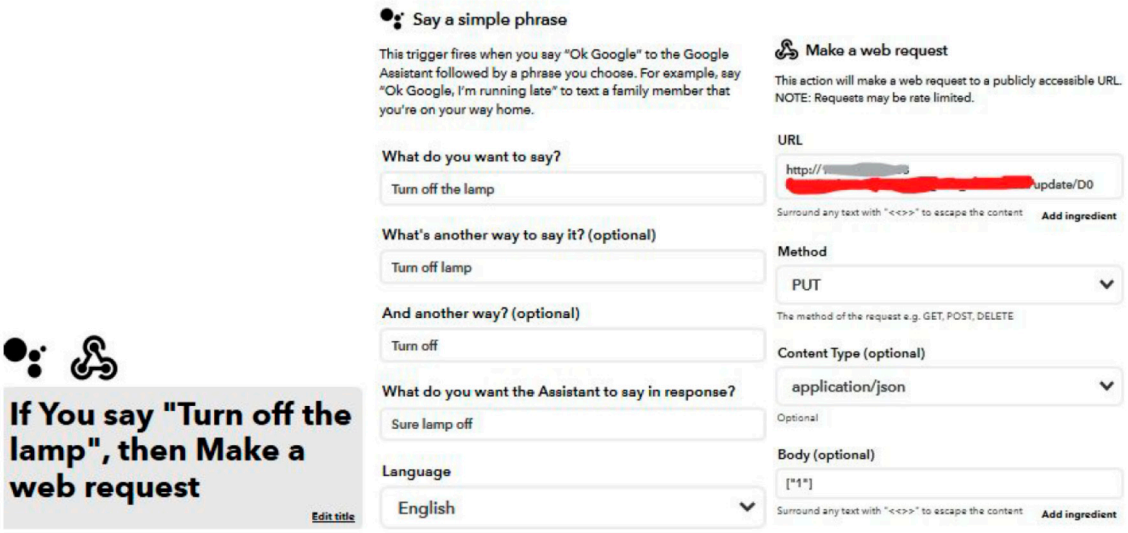

Figure 6. IFTTT trigger turn off.

3. Proposed Work

In this project, it is aimed to communicate Google Home Mini with an IoT device safely. As an IoT device, a so-called smart lamp with Wi-Fi connection is preferred. The user can switch this lamp on and off with the device control application on his android-based phone. He can also control the lamp by sending voice commands to Google Home Mini. All systems can be seen in Figure 7.

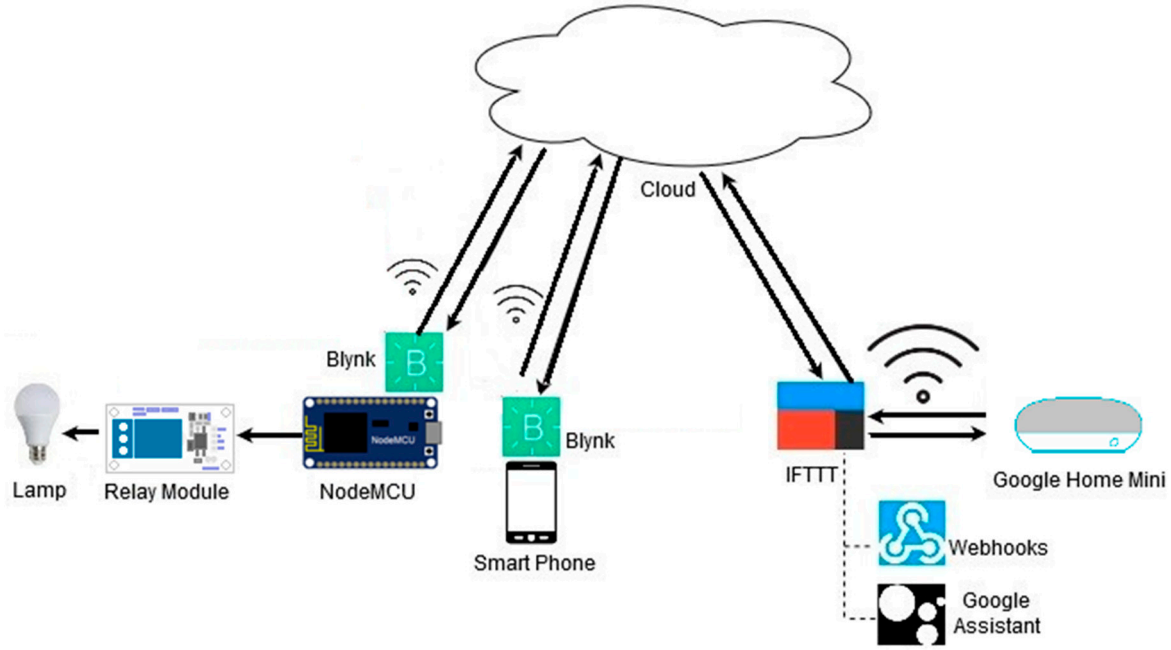

Figure 7. IoT device system.

The project uses Node MCU ESP8266 that works like a small-scale web server. After the hardware part is completed, a control application is made for the smart lamp. The Blynk application is used for this. Then, a new application for the project is created from

https://www.ifttt.com (accessed on 28 August 2020). Google Assistant is selected, and the trigger is generated.

With NodeMCU, the Internet connection of the lamp is provided. Thus, the lamp becomes an IoT device with an Internet connection, which is recognized by other devices in the network. The bulb used in this project is a 220-volt bulb. However, the prepared system is a 5-volt system. Therefore, a switch is needed to control the lamp at low voltage. A relay module is also used in this section. After installing the required codes, a 5-volt adapter is needed to use the ready-to-operate NodeMCU. Female-to-female jumper cables are used to provide the connection between NodeMCU and relay module pins. An android-based smartphone is used for the applications of the hardware used in the project. The smartphone has Android 8.0.0. Arduino IDE has been preferred as the development environment for writing NodeMCU codes. There are also other development environments that can be used besides Arduino IDE. The computer used in the project has a Windows 10 operating system. Google Home Mini, a smart home assistant, was chosen for voice control and IoT device control. The reason for choosing Google Home Mini instead of Google Home is that it has enough features for IoT device management. Google Home can also be preferred.

After the hardware part of the project was completed, some analyses were conducted to determine whether the communication between the IoT device and Google Home Mini was secure. Firstly, the incoming and outgoing packets in this system using Wi-Fi and cloud technologies should be observed. Many network traffic analysis software programs are available for these observations. In this project, Wireshark was used as the network traffic analysis software program. This is a free program. The computer on which the program is installed is a 64-bit computer with Windows 10 operating system. Google Home Mini and an IoT device are installed, with the system running after the Wireshark program is opened and the network traffic is examined. When the smart lamp is turned on and off by sending voice commands to the Google Home Mini, the program displays detailed information about incoming–outgoing packets, device IP addresses, protocols used, and packets.

Before starting a secure communication analysis, the security risks of IoT devices were first investigated to select a specific study area. IoT devices utilize time information to communicate with other devices on the network without problems. It uses the time information for the synchronization process. However, these devices cannot store time information themselves, so they need to get them from an outside source. This can cause some security risks. As the subject of the analysis, the process of obtaining the time information of the IoT devices from an external source is examined, as shown in Figure 8.

The IoT device can communicate with the Google Home Mini via the application prepared on the Blynk platform. In this case, the libraries of the Blynk application were examined first, and it was determined where the application obtained the time information. There is a file named BlynkSimpleEsp8266_SSL_h in the Blynk libraries of the Arduino project where NodeMCU codes are written. This file contains a function called ‘connect’. This function is shown in Figure 9.

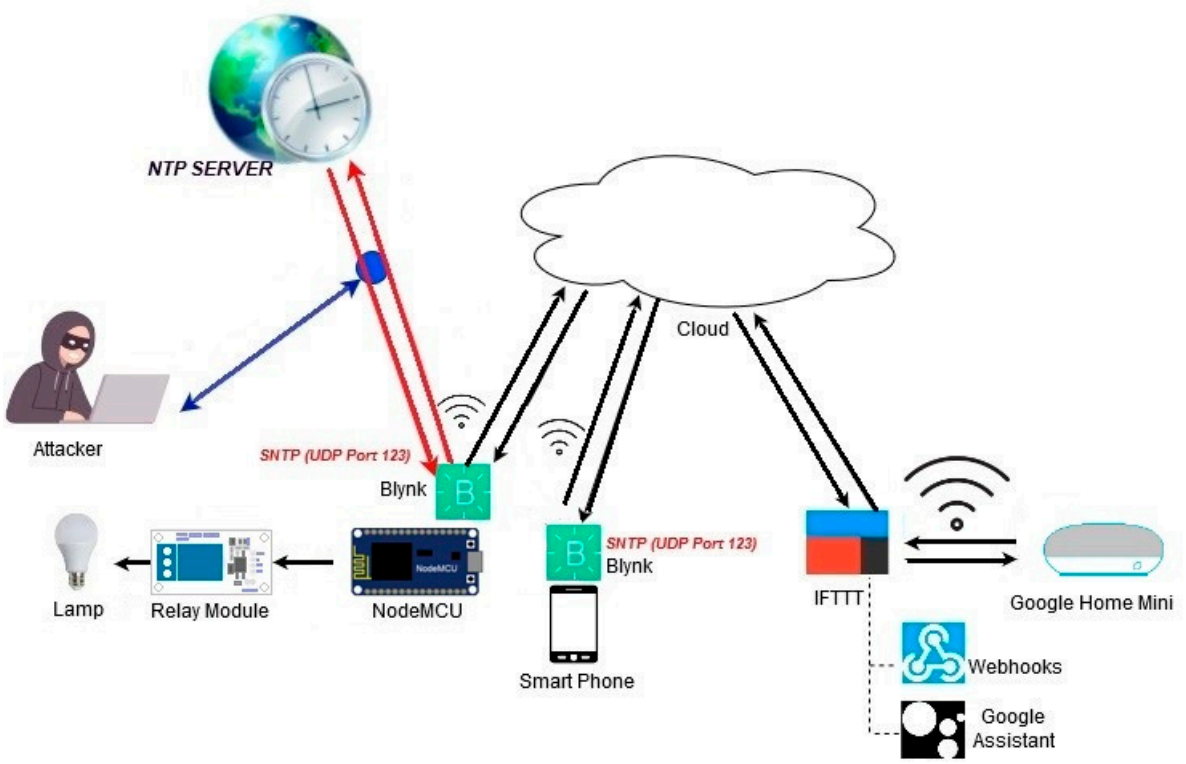

Figure 8. Attack scenario to the connection between Google Home Mini and NodeMCU.

```
configTime(0, 0, "pool.ntp.org", "time.nist.gov");
```

Figure 9. configTime function.

The application receives time information from the uniform resource locators (URLs) of the servers shown in Figure 9. Time synchronization is provided by using simple network time protocol (SNTP). This process is used to verify the validity of the transport layer security (TLS) certificate that is provided by the server. As a result of some investigations, security vulnerabilities were found on this subject. There have been some attacks on NTP. One of these attacks is the on-path time-shifting attack. The attacker could make some changes to the time information. Changes can be made in the time part or even the year part of the time information. In the other attack, domain name system security extensions (DNSSEC) was attacked. DNSSEC provides encrypted verification of DNS data. As a result of this attack over NTP, the encryption key and signature of the DNSSEC expired [12]. Based on these studies, the packets using TLS protocol in the network traffic that was monitored with Wireshark were examined.

Figure 10 shows the packets using the TLS protocol in the traffic generated by the communication between the IoT device and the Google Home Mini. A handshake is performed here. Details of these packages have been examined. However, no time information was observed that was not encrypted. Another review was made for the packets using multicast domain name system (MDNS) protocol.

```
32 33.178213    192.168.1.31    52.114.76.35    TLSv1.2  262 Client Hello
37 33.383940    52.114.76.35    192.168.1.31    TLSv1.2 1260 Server Hello, Certificate, Server Key Exchange, Server Hello Done
38 33.385268    192.168.1.31    52.114.76.35    TLSv1.2  147 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
39 33.474470    52.114.76.35    192.168.1.31    TLSv1.2  105 Change Cipher Spec, Encrypted Handshake Message
40 33.482580    192.168.1.31    52.114.76.35    TLSv1.2 1024 Application Data
41 33.482703    192.168.1.31    52.114.76.35    TLSv1.2  911 Application Data
43 33.622072    52.114.76.35    192.168.1.31    TLSv1.2  413 Application Data
```

Figure 10. Packages that use the transport layer security (TLS) protocol.

The packages shown in Figure 11 belong to the Google Home Mini and the smartphone where the Blynk application is located. When the packages were examined, it was seen that Google Home Mini sends a query and receives an answer in return, and then, the smartphone packages are received. When these packages are examined in detail, no time information was found that was not encrypted.

20	18.642019	192.168.1.20	224.0.0.251	MDNS	82	Standard query 0x0000 PTR _googlezone._tcp.local, "QM"
21	18.642020	192.168.1.20	224.0.0.251	MDNS	268	Standard query response 0x0000 PTR f2415b87-14ea-0658-
22	19.908203	192.168.1.22	224.0.0.251	MDNS	152	Standard query 0x0007 PTR _%9E5E7C8F47989526C9BCD95D24
23	19.917441	192.168.1.20	224.0.0.251	MDNS	416	Standard query response 0x0000 PTR Google-Home-Mini-f2
24	19.917663	192.168.1.20	224.0.0.251	MDNS	401	Standard query response 0x0000 PTR Google-Home-Mini-f2

Figure 11. Packages that use the multicast domain name system (MDNS) protocol.

4. Results and Discussion

With the widespread use of IoT devices, the usage areas of the IoT devices are also differentiated. Smart homes are built by communicating with some devices such as Google Home Mini and IoT devices over the cloud using wireless connections. Many critical tasks in these homes are fulfilled by IoT devices. During the wireless connection of these devices, there are possibilities such as an attacker interrupting communication, changing the transmitted information, or stopping the system by using other attack methods. For this reason, establishing a secure system has great importance.

In this study, a test has been made for the security of the connection between the Google Home Mini device and an IoT device that can also be controlled by an application on a mobile phone. Blynk application uses Secure Sockets Layer(SSL) for secure connection. The Blynk application provides date and time information from a server with SNTP. This information reaches the IoT device over the network. The attacker may attempt to capture or change this information. For this reason, the data flowing through the UDP 123 port were listened to on the wireless network. Access to the SNTP server and the information sent and received were monitored. In this way, if this information is transferred plaintext in traffic, it is planned to make an attack attempt.

As a result of the tests, the security risk on the SNTP server could not detected. Attack attempts related to this study are ongoing, and the results of this study will be shared in the future if relevant data and results are reached. The study guides future studies on this subject. Step-by-step procedures and tests are shared. By using the data obtained and shared in this study, researchers can diversify and make the tests more comprehensive. Apart from the tried method regarding SNTP, it is possible to attack with different methods.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable

Data Availability Statement: The study did not report data so it can be excluded.

References

1. Keoh, S.L.; Kumar, S.S.; Tschofenig, H. Securing the internet of things: A standardization perspective. *IEEE Internet Things J.* **2014**, *1*, 265–275.
2. Notra, S.; Siddiqi, M.; Gharakheili, H.H.; Sivaraman, V.; Boreli, R. An experimental study of security and privacy risks with emerging household appliances. In Proceedings of the 2014 IEEE Conference on Communications and Network Security, San Francisco, CA, USA, 29–31 October 2014; pp. 79–84.
3. Zhang, N.; Mi, X.; Feng, X.; Wang, X.; Tian, Y.; Qian, F. Understanding and mitigating the security risks of voice-controlled third-party skills on amazon alexa and google home. *arXiv* **2018**, arXiv:1805.01525.
4. Baruah, B.; Dhal, S. A two-factor authentication scheme against FDM attack in IFTTT based Smart Home System. *Comput. Secur.* **2018**, *77*, 21–35.
5. Fernandes, E.; Rahmati, A.; Jung, J.; Prakash, A. Decoupled-ifttt: Constraining privilege in trigger-action platforms for the internet of things. *arXiv* **2017**, arXiv:1707.00405.
6. Zeroday. A lua Based Firmware for Wifi-Soc esp8266. Github. 2015. Available online: https://github.com/nodemcu/nodemcu-firmware (accessed on 26 October 2020).

7. Wiguna, H. NodeMCU LUA Firmware. Hackaday. 2015. Available online: https://hackaday.io/project/3465-playing-with-esp8266/log/11449-nodemcu-lua-firmware (accessed on 26 October 2020).
8. SunFounder. 2 Channel 5 V Relay Module. 2017. Available online: http://wiki.sunfounder.cc/index.php?title=2_Channel_5V_Relay_Module (accessed on 26 October 2020).
9. Google Home Specifications. Google Home Help. Google. 2017. Available online: https://support.google.com/googlenest/answer/7072284 (accessed on 28 August 2020).
10. Home, Blynk. Available online: http://www.blynk.cc/ (accessed on 26 October 2020).
11. IFTTT. Helps Everything Work Better Together. Available online: https://ifttt.com/ (accessed on 26 October 2020).
12. Malhotra, A.; Cohen, I.E.; Brakke, E.; Goldberg, S. Attacking the Network Time Protocol. In Proceedings of the NDSS, 21-24 February 2016, San Diego, California; doi:10.14722/ndss.2016.23090.