Threat modeling and analysis of voice assistant applications

Geumhwan Cho[1], Jusop Choi[1], Hyoungshick Kim[1], Sangwon Hyun[2], and Jungwoo Ryoo[3]

[1] Sungkyunkwan University, South Korea
[2] Chosun University, South Korea
[3] Pennsylvania State University, USA
{geumhwan, cjs1992, hyoung}@skku.edu, shyun@chosun.ac.kr, jryoo@psu.edu

Abstract. Voice assistant is an application that helps users to interact with their devices using voice commands in a more intuitive and natural manner. Recently, many voice assistant applications have been popularly deployed on smartphones and voice-controlled smart speakers. However, the threat and security of those applications have been examined only in very few studies. In this paper, we identify potential threats to voice assistant applications and assess the risk of those threats using the STRIDE and DREAD models. Our threat modeling demonstrates that generic voice assistants can potentially have 16 security threats. To mitigate the identified threats, we also discuss several defense strategies.

Keywords: Voice assistant · Threat modeling · STRIDE · DREAD

1 Introduction

Voice assistant is a software program that helps users to interact with services (e.g., search engine) and applications (phone application) using voice commands with a more intuitive and convenient user interface mechanism. In general, voice assistant application runs as a background process and can be activated by using a reserved voice command (e.g., "Hey, Siri" and "Alexa"). Popular voice assistants including Siri (Apple), Alexa (Amazon) and Now (Google) help people shop online, send instant messages, and make phone calls, all through voice commands. However, to our knowledge there is no study analyzing security threats to voice assistants through a threat modeling process. Only a few studies experimentally demonstrated that commercial voice assistant applications are vulnerable to various forms of voice presentation attacks (e.g., [4, 11]).

The goal of this paper is to identify potentially serious security threats to voice assistants and suggest several mitigation techniques to mitigate them. We first identify what threats exist and how risky the threats are by using the Security Development Lifecycle (SDL) threat modeling tool [10] that systemically analyzes threats based on the data flow diagrams of a target system. The tool has been widely used for identifying security threats and analyzing corresponding security requirements. The key contributions of this paper are as follows:

- We provide a security analysis based on the SDL threat modeling methodology. We describe how a generic voice assistant application works with a data flow diagram. We then use the STRIDE approach [10] for categorizing 16 identified threats and the DREAD model [2] for assessing the risk of the threats (read Section 3).
- We discuss three possible attack scenarios that could lead to severe damages to systems using voice assistant applications (read Section 4) and suggest several defense mechanisms to mitigate those threats (read Section 5).

2 Background

2.1 Voice assistant

Voice assistants have become more widely used for many purposes (e.g., playing music, setting timers and getting weather forecasts). Figure 1 shows an example architecture of voice assistant systems. A user initiates a voice assistant by issuing a voice command. For example, a user says, "*what time is it?*" and the voice assistant delivers the user's voice stream that the user requested to a voice assistant server. Next, the server interprets the voice stream and then requests the corresponding service to the cloud. The response to the user's voice command is delivered in the reverse order of the service request process. Consequently, the user can obtain the response of requested command.

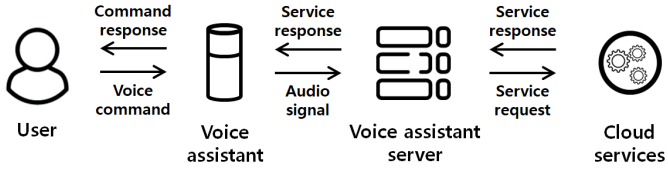

Fig. 1. A generic architecture of voice assistant systems.

2.2 Threat modeling

Threat modeling is a process to identify potential threats to a system and evaluate the risk levels of identified threats. This process is helpful for reducing the risk from threats in the target system. In this paper, we use a threat modeling tool [10] to identify threats to voice assistant systems in a more systematic

STRIDE is a threat classification model developed by Microsoft. STRIDE is an acronym containing the following concepts [10]. Our primary goal in this paper is to identify and categorize threats against voice assistant systems from the attacker's point of view, and the STRIDE model fits this goal.

- **S**poofing is an attempt to gain access to a system using a forged identity.

- **T**ampering is data corruption during network communication.
- **R**epudiation is a user's refusal to acknowledge participation in a transaction.
- **I**nformation disclosure is the unwanted exposure and loss of private data's confidentiality.
- **D**enial of Service (DoS) is an attack against system availability.
- **E**levation of privileges is an attempt to raise the privilege level of users by exploiting vulnerabilities.

DREAD is mainly used for quantifying the level of risks caused by threats [2]. In this paper, we used the DREAD model because it is especially useful to rank and prioritize threats according to their severity.s Using the DREAD model, we can quantify the severity of each threat with numeric values (0, 5 and 10) assigned to each of the five categories described as follows [1], and consequently identify the threats that need to be dealt with higher priorities.s

- **Damage Potential** measures the extent of the possible damage incurred by a threat. If the attacker could damage the entire system and data by exploiting a vulnerability, it would be the worst (10).
- **Reproducibility** measures how easy the attack or threat can be repeated.
- **Exploitability** is a metric that quantifies how much effort is required to launch an attack. If anyone can launch an attack, it would be the worst (10).
- **Affected Users** captures how many people would be affected if the attack was launched. It is usually a measure of what percentage of users are affected.
- **Discoverability** is a metric that indicates how easy a threat can be detected. If an attack is easily identifiable, it would be 10.

3 Security analysis

3.1 Data Flow Diagram (DFD)

To identify security threats, we first draw a DFD (see Figure 2) with the eight entities using the Microsoft's threat modeling tool [10]. `Human user` is an entity who uses the `Voice assistant application (app)` and controls `IoT devices` through the app. The voice assistant is working with a `Voice assistant server` to process a user's voice commands. The server typically converts the user's voice command to a service request message and sends it to an appropriate cloud server that can provide the service requested by the user. In addition, it is especially important to obtain a DFD that reflects the procedures of real-world voice assistant systems for practical modeling of threats. The DFD shown in Figure 2 has been constructed through our real development process, and has a strong similarity with popular voice assistant systems (e.g., Siri and Alexa).

The SDL threat modeling tool analyzed the DFD and automatically identified a list of 36 potential threats based on the STRIDE categories. We carefully examined the feasibility of attacks exploiting those vulnerabilities, identified that 20 of the identified threats are unrealistic in real-world settings, and finally selected 16 threats as valid ones by excluding the unrealistic threats. For example,

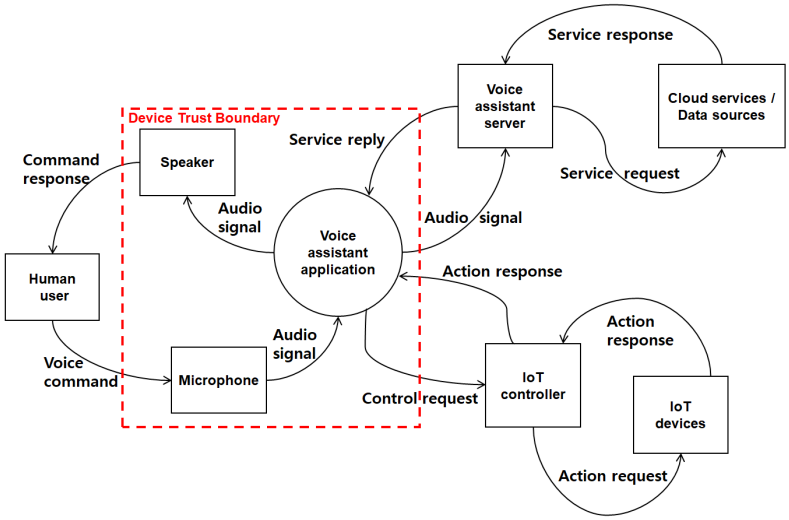

Fig. 2. DFD for generic voice assistant systems. The red rectangle indicates within voice assistant device.

in Figure 2, the spoofing attack on voice command to `Microphone` has been proved as a feasible attack by some previous studies [4, 11], thus we categorized it as a valid threat. As another example, on the other hand, the spoofing attack on command response to `Human user` is categorized as an invalid one, because it is unrealistic to assume that an attacker is able to put a fake speaker or physically compromise a victim's device without being detected by the victim.

3.2 Security threat analysis

To identify the security threats of the voice assistant system, we focus on analyzing threats associated with the following inbound/outbound data flows to/from the device (denoted as `Device trust boundary`) where `Speaker`, `Microphone` and `Voice assistant app` are placed; voice command, audio signal, service reply, control request, action response, and command response in Figure 2. We do not consider the data flows inside the device trust boundary since we assume no physical attack against the voice assistant device. In addition, we ignore the data flows between `IoT controller` and `IoT devices` and between `Voice assistant server` and `Cloud services` in this analysis, which are not directly related with the voice assistant device. In the following, we will discuss 16 possible threats associated with the six data flows mentioned above.

Voice command to Microphone. `Human user` issues a voice command to activate the `Voice assistant app` and the voice command first arrives at `Microphone`. We found four possible threats between `Human user` and `Microphone`.

- **Spoofing:** An attacker may attempt to spoof `Microphone` by impersonating a legitimate human user's voice. As a possible implementation, the attacker can simply record a victim's voices and replay them.
- **Tampering:** A voice command issued by `Human user` could be captured and modified by the attacker. Due to the broadcast nature of sound signals, anyone can hear and record the voice command. The attacker can then modify the recorded voice signals to generate the malformed voice command that causes unauthorized operations to be performed
- **Denial of Service:** The role of `Microphone` is to receive the voice command from `Human User` and deliver it to `Voice assistant app`. The attacker can possibly launch a DoS attack by continuously injecting attack sounds to interfere with the normal voice command from the legitimate `Human User`. To secretly launch such attacks, the attacker might use hidden [4] and/or inaudible sounds [11] that a human cannot recognize and/or hear.

Audio signal to Voice assistant server. `Voice assistant app` makes a request to analyze the audio signal (voice command) received from `Microphone`. Between `Voice assistant app` and `Voice assistant server`, spoofing attacks could be launched.

- **Spoofing:** The spoofing attack can be launched against `Voice assistant server` if there is no authentication mechanism. By impersonating a legitimate `Voice assistant app`, an attacker can transmit *unauthorized* audio signals to `Voice assistant server`. Consequently, the attacker can access the voice assistant service in an unauthorized manner.

Service reply to voice assistant app. `Voice assistant server` sends a reply to `Voice assistant app`. From the reply, `Voice assistant app` decides what operation to perform. We identified four possible threats on this data flow as follows.

- **Tampering:** The service reply might be tampered by the attacker if there is no guarantee on the integrity of the service reply. That is, the attacker can capture a service reply and modify it to deceive `Voice assistant app`. Consequently, the attacker can deliver the malformed service reply to cause unauthorized operations on `Voice assistant app` or distribute unwanted information (e.g., advertisements) to `Voice assistant app`.
- **Information Disclosure:** The attacker might launch a sniffing attack of the service reply if there is no confidentiality protection of the service reply. For example, if the service reply is not encrypted, the attacker can extract some privacy sensitive information by eavesdropping the service reply.
- **Denial of Service:** The attacker could perform a DoS attack to disrupt the availability of `Voice assistant app`. The attacker has two options of DoS attacks. The attacker generates a service reply (with tampering attack) to contain malformed commands that might compromise the availability

of `Voice assistant app`. The other option is to send a massive amount of service replies and consequently cause congestion on `Voice assistant app`.
– **Elevation of Privileges:** The attacker can generate malformed service replies containing malicious commands to exploit some vulnerabilities (e.g., vulnerable functions, insecure administrator password, etc.) on `Voice assistant app`. The execution of these malicious commands may allow the attacker to get a higher privilege than what is normally given by `Voice assistant app`.

Control request to IoT controller. Control requests are generated when `Human user` tries to control `IoT devices` with voice commands. We identified a threat that allows an attacker to spoof `IoT controller` with forged requests.

– **Spoofing:** A spoofing attack against `IoT controller` might be possible if there is no authentication mechanism between `Voice assistant app` and `IoT controller`. By impersonating `Voice assistant app`, the attacker can inject malicious control requests to `IoT controller`, and these forged control requests eventually allow the attacker to control security critical `IoT devices` (e.g., a digital door lock).
– **Tampering:** A tampering attack against `IoT controller` is possible if there is no integrity protection of control requests. Modifying control requests is to perform unauthorized operations on `IoT devices` (e.g., unlocking the door and forcibly turning on the fire alarm).

Action response to Voice assistant app. `IoT controller` responds `Voice assistant app` with an action response generated by `IoT devices`. The action response contains some information collected by `IoT devices`, such as the current temperature in the user's home, and also the status of `IoT devices`. We will explain six possible threats on the action response between `IoT controller` and `Voice assistant app`.

– **Spoofing:** `Voice assistant app` might be spoofed by an attacker if there is no authentication mechanism between `Voice assistant app` and `IoT controller`. The attacker can send an action response to `Voice assistant app` by impersonating `IoT controller`. As a result, the attacker can send any action response to `Voice assistant app`.
– **Tampering:** Without a proper integrity protection of an action response, an attacker can intentionally modify any captured action response and inject the forged action response to `Voice assistant app`.
– **Information Disclosure:** A sniffing attack against an action response is possible if there is no confidentiality protection of an action response. If an action response is transmitted without encryption, the attacker can easily obtain sensitive information from a captured action response.
– **Denial of Service:** The attacker can inject malformed action responses with some malicious commands whose execution causes intentional faults on `Voice assistant app`. In addition, the attacker can cause congestion on

`Voice assistant app` by injecting an excessive amount of action responses. As a result, such DoS attacks can seriously disrupt normal operations of `Voice assistant app`.
- **Elevation of Privilege:** The action response could be misused by an attacker for illegal privilege escalation. For this attack, the attacker can generate an action response with malicious commands to exploit some vulnerabilities (e.g., vulnerable functions, insecure administrator password, etc.) and inject the action response to `Voice assistant app`. The execution of these malicious commands possibly allows the attacker to get a higher privilege than what is normally given by `Voice assistant app`.

Command response to human user. `Human user` obtains a service response via `Speaker`, which is the result of the voice command issued by `Human user`. We found that some sensitive information might be exposed to those who are physically close to `Speaker` because the command responses are usually broadcasted over air.

- **Information Disclosure:** `Speaker` simply broadcasts a service response over the air. Thus if the service response contains some sensitive information, anyone within the audible range can hear and/or record the service response. For this reason, `Human user` who wants to receive a command response from `Speaker` should carefully run `Voice assistant app`.

3.3 Risk analysis

This section explains how we can evaluate the risk of each threat identified in Section 3.2 based on the DREAD assessment model [10, 6]. We focus on assessing how effective attacks are and how easy they are to launch. Each DREAD rating is the average of 5 categories and calculated using $(D + R + E + A + D)/5$.

We calculated the risk score (0, 5, and 10) of each of the 16 threats based on the DREAD model (see Table 1) and categorized the 16 threats into the following 3 orders of priority according to their risk scores: low (0 to 3), medium (4 to 7) and high (over 8). In summary, the 16 threats were categorized as follows: 2 threats in the low, 10 threats in the medium, and 4 threats in the high. In the following, we will describe the detailed procedure in which risk scores are calculated for several examples of threats (spoofing and DoS attack against `Microphone`, information disclosure of service reply).

Spoofing attack against microphone. The open nature of the voice channels and the lack of authentication mechanisms make the voice assistants vulnerable to spoofing attacks such as voice replay attacks and voice impersonation attacks. As a result of spoofing attacks, an attacker can gain legitimate users' privileges (Damage Potential = 10). To launch such spoofing attack, the attacker generally require simple tools (e.g., recorders, speakers) without requiring sophisticated skills (Exploitability = 10), and this makes the attacks easy to launch and

Table 1. Risk assessment of voice assistant.

Data flow	Threat	D	R	E	A	D	Average
Voice command	Spoofing on `Microphone`	10	10	10	10	10	**10**
	Tampering with voice command	10	0	0	10	0	**4**
	DoS to `Microphone`	0	10	0	10	0	**4**
Audio signal	Spoofing on `Voice assistant server`	0	0	10	5	0	**3**
Service reply	Tampering with service reply	10	10	10	10	5	**9**
	Information disclosure of service reply	10	10	10	10	0	**8**
	DoS to `Voice assistant app`	0	10	5	10	10	**7**
	Elevation of privilege into `Voice assistant app`	10	10	0	10	0	**6**
Control request	Spoofing on `IoT controller`	10	10	5	10	0	**7**
	Tampering with control request	10	10	5	10	10	**9**
Action response	Spoofing on `IoT controller`	0	10	5	10	10	**7**
	Tampering with action response	0	10	0	10	10	**6**
	Information disclosure of action response	10	10	5	5	0	**6**
	DoS to `Voice assistant app`	0	10	10	10	10	**10**
	Elevation of privilege into `Voice assistant app`	10	0	0	0	0	**2**
Command response	Information disclosure of command response	0	10	10	10	0	**6**

reproduce (Reproducibility = 10). Most voice assistants are focusing on text-to-speech translation and service provisioning without paying much attention to security (Discoverability = 10), thus the impacts of these types of attacks are significant (Affected Users = 10). Taking all these aspects into consideration, the DREAD rating is set to 10 and its priority is set to high.

DoS attack against microphone. An attacker can launch DoS attacks against `Microphone` to disrupt the normal operations of voice assistants. Such DoS attacks interrupt legitimate users from using the voice assistant normally, but they do not directly damage the system and data (Damage Potential = 0). A simple form of DoS attack is playing audio files of attack sounds via a speaker, and this type of attack can not only be reproduced easily (Reproducibility =10) but also affect most voice assistant users (Affected Users = 10). On the other hand, other types of DoS attack are not easy to launch because they require some specialized skills such as generating inaudible sounds (Exploitability = 0). In the case of using inaudible sounds, it is difficult for users to detect the attacks (Discoverability = 0). As a result, the DREAD rating is calculated as 4 and its priority is set to medium.

Information disclosure of service reply. Service replies from `Voice assistant server` may be sniffed by an attacker. Depending on the types of data an attacker sniffs, it may be used to attack other parts of the system. Information disclosure can cause both direct and indirect damages to the system and data assets of users (Damage Potential = 10). With the assumption that service replies are not encrypted, sniffing service replies can be repeated whenever an attacker

wants (Reproducibility = 10). In addition, sniffing attacks only require simple tools, and this makes these attacks easy to launch (Exploitability = 10). `Voice assistant server` is generally connected with a lot of users to provide them with services, and sniffing service replies from `Voice assistant server` can affect the privacy of all the users (Affected Users = 10). Due to the nature of information leakage, users do not know that their information has been leaked until the information is illegally used, and this makes it difficult to detect such sniffing attacks (Discoverability = 0).

4 Attack scenarios

4.1 Spoofing against microphone scenario

The open nature of voice channels and the lack of authentication mechanisms make voice assistants vulnerable to spoofing attacks such as voice replay attacks and voice impersonation attacks. Although audio microphones and speakers are enough to launch spoofing attacks, other sophisticated techniques such as signal processing can also be used to further improve the effectiveness of the attacks. For example, an attacker can use the so-called sound mosaic technique [8] to generate malicious voice commands from voice sound samples of victims, collected in advance. Specifically, the sound mosaic technique allows the attacker to generate malicious voice commands by dividing and concatenating victims' voice samples. This technique is especially useful for impersonating famous people whose voice samples can be gathered easily.

Most of voice assistants do not provide voice authentication, that is, they do not care about who issues voice commands. This inherently makes them vulnerable to attacks injecting malicious voice commands. Although voice assistant devices are usually placed in the proximity of users, such attacks are still possible in this environment without raising the victim's attention, by injecting malicious voice commands that are only recognizable by machines but not by humans [4, 11]. For example, an attacker can secretly embed inaudible malicious voice commands into a television broadcast or an announcement on elevators so that the malicious voice commands affect voice assistants in a stealthy manner.

4.2 DoS against microphone scenario

Because the voice sounds are easily influenced by their surrounding environments, the voice assistant on a smartphone does not work well near a busy road or around a construction site. Therefore, it is possible to generate a sound which influences the voice assistant. DoS attack usually occurs in a network, and it is mitigated by blocking the packets or limiting the service requests. Unlike detecting traditional DoS attacks, the attack against the voice assistants is hard to recognize and prevent. Especially, if the attacker uses inaudible sounds for the DoS attacks, it is very hard for users to discover the attack occurring.

4.3 Information disclosure of service reply scenario

The service reply might contain a variety of information including the user's condition, what the user's interests are, some sensitive data and and so on. The attacker can obtain much information by sniffing the communication channel. This leaked information can be used in social engineering attacks or phishing attacks. To mitigate this threat, packet encryption is the best strategy but the attacker may still infer some sensitive information from encrypted packets when encryption schemes are insecurely implemented. For example, traffic analysis (e.g., [7]) can be applied to obtain some sensitive information even when network packets are encrypted. Thus, the developer should carefully implement a secure encryption algorithm with a proper encryption mode and a padding scheme.

5 Recommendations

In this section, we offer some practical recommendations to mitigate potentially serious security scenarios described in Section 4.

To mitigate the spoofing attack in Section 4.1, destination authentication and liveness detection in data flows are required. That is, the voice assistant has to identify who the voice's owner is and whether the voice is generated in the present, not the past. One of the voice owner identification methods is the voiceprint analysis [5]. A voiceprint is the characteristics of a human voice, and the characteristics are unique. In the cloud computing, the accuracy of the voiceprint authentication achieved 3.2% in FRR (False Reject Rate) [13].

The voiceprint method is vulnerable to recorded voice. Therefore, voice assistants have to use the liveness detection method. The liveness detection calculates phoneme localization with two microphones. Through the liveness detection method, the voice assistant can distinguish whether the input voice is from the user in real time or replayed. Zhang et al. [12] achieved over 99% accuracy and under 1% EER (Equal Error Rate).

To mitigate the denial of service attack of Section 4.2, limiting availability in data flows is required. That is, the voice assistant has to input the audible hertz sounds (e.g., 20–20,000 Hz) using signal processing for low pass filter. If the voice assistant only accepts the human audible sound ranging from 20 to 20,000 Hz, it is easier to detect anomalies. DoS attacks can be mitigated by limiting the number of the input data flows and filtering malformed data.

The simplest way to prevent the information disclosure in Section 4.3 is the encryption of data flows, such as SSL/TLS network protocols. However, imperfect encryption can still lead to information leakages. Many Android apps use SSL/TLS to protect their sensitive information. However, the sensitive data protected by SSL/TLS was vulnerable to Man-in-the-Middle attacks because some developers often used incorrect options or implementations [9]. In addition, the HTTPS protocol which is popularly used in protecting home banking, e-commerce, and e-procurement was vulnerable to Man-in-the-Middle attacks when the attacker had access to the victim's network [3].

Table 2. Suggested mitigation according to STRIDE category.

Data flow	Threat and Mitigation
Voice command	**Spoofing on microphone:** Identify the authenticity of the voice command (e.g., voice recognition, voice liveness detection).
	Tampering with voice command: Provide a detection mechanism for distinguishing legitimate users' voice commands from forged ones.
	DoS to microphone: Deploy low-pass filter to cut off higher frequency than audio frequency.
Audio signal	**Spoofing on voice assistant server:** Identify the legitimate voice assistant application using authentication mechanism.
Service reply	**Tampering with service reply:** Protect the integrity of the service reply (e.g., HMAC), or use a secure channel.
	Information disclosure of service reply: Encrypt the service reply (e.g., TLS), or use a secure channel.
	DoS to voice assistant application: Limit the number of the service reply, or use a filter to distinguish malformed response or command.
	Elevation of privilege into voice assistant application: Check the validity of the input data (e.g., length of the input data).
Control request	**Spoofing on IoT controller:** Identify the legitimate voice assistant application using authentication mechanism.
	Tampering with control request: Protect the integrity of control requests (e.g., HMAC), or use a secure channel.
Action response	**Spoofing on IoT controller:** Identify the legitimate voice assistant application using authentication mechanism.
	Tampering with action response: Provide the integrity of action response (e.g., HMAC), or use secure channel.
	Information disclosure of action response: Encrypt the action response (e.g., TLS), or use a secure channel.
	DoS to voice assistant application: Limit the number of action response, or use a filter to distinguish malformed response.
	Elevation of privilege into voice assistant app: Check the validity of the input data (e.g., check the length of the input data) and use secure function within voice assistant application.
Command response	**Information disclosure of command response:** Avoid using command responses containing sensitive information about the user.

The other threats can also be handled by a variety of mitigation methods. According to a given situation and capabilities, the mitigation methods should be properly chosen and implemented. Unsurprisingly, mitigation methods are not secure forever. Therefore, threat modeling analysts should periodically analyze possible threats and prepare proper strategies to mitigate those threats.

6 Conclusion

This paper analyzed potential security threats for voice assistant systems and suggested several mitigation strategies to reduces the risk of the identified threats through threat modeling analysis that has been widely used in the field of information security. Based on the threat analysis results, we also presented several attack scenarios. Finally, we proposed several practical defense strategies to

mitigate the identified threats. In future work, we will implement the discovered attacks against real-world voice assistant systems to show their feasibility.

Acknowledgments. This work was supported in part by the ITRC (IITP-2018-2015-0-00403) and the NRF (No.2017K1A3A1A17092614). The authors would like to thank all the anonymous reviewers for their valuable feedback.

References

1. Anand, P., Ryoo, J., Kim, H., Kim, E.: Threat Assessment in the Cloud Environment: A Quantitative Approach for Security Pattern Selection. In: Proceedings of the 10th ACM International Conference on Ubiquitous Information Management and Communication (2016)
2. Burns, S.F.: Threat Modeling: A Process To Ensure Application Security. GIAC Security Essentials Certification (GSEC) Practical Assignment (2005)
3. Callegati, F., Cerroni, W., Ramilli, M.: Man-in-the-Middle Attack to the HTTPS Protocol. IEEE Security & Privacy (2009)
4. Carlini, N., Mishra, P., Vaidya, T., Zhang, Y., Sherr, M., Shields, C., Wagner, D., Zhou, W.: Hidden Voice Commands. In: Proceedings of the 25th USENIX Security Symposium (2016)
5. Garcia-Salicetti, S., Beumier, C., Chollet, G., Dorizzi, B., Les Jardins, J.L., Lunter, J., Ni, Y., Petrovska-Delacrétaz, D.: BIOMET: A Multimodal Person Authentication Database Including Face, Voice, Fingerprint, Hand and Signature Modalities. In: Proceedings of the 4th International Conference on Audio-and Video-based Biometric Person Authentication (2003)
6. Meier, J., Mackman, A., Dunner, M., Vasireddy, S., Escamilla, R., Murukan, A.: Improving web application security: threats and countermeasures. Microsoft Corporation (2003)
7. Park, K., Kim, H.: Encryption Is Not Enough: Inferring user activities on KakaoTalk with traffic analysis. In: Proceedings of the 16th International Workshop on Information Security Applications (2015)
8. Shih, T.K., Tang, N.C., Tsai, J.C., Hwang, J.N.: Video motion interpolation for special effect applications. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) (2011)
9. Sounthiraraj, D., Sahs, J., Greenwood, G., Lin, Z., Khan, L.: Smv-hunter: Large scale, automated detection of ssl/tls man-in-the-middle vulnerabilities in android apps. In: Proceedings of the 21st Annual Network and Distributed System Security Symposium (2014)
10. Swiderski, F., Snyder, W.: Threat Modeling (Microsoft Professional), vol. 7. Microsoft Press (2004)
11. Zhang, G., Yan, C., Ji, X., Zhang, T., Zhang, T., Xu, W.: DolphinAttack: Inaudible voice commands. In: Proceedings of the 24th ACM SIGSAC Conference on Computer and Communications Security (2017)
12. Zhang, L., Tan, S., Yang, J., Chen, Y.: Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In: Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (2016)
13. Zhu, H.H., He, Q.H., Tang, H., Cao, W.H.: Voiceprint-biometric template design and authentication based on cloud computing security. In: Proceedings of 4th IEEE International Conference on Cloud and Service Computing (2011)