Threat Modeling: Detecting Security Threats in an Automated Teller Machine (ATM)

A Report by

Mark Kardash

SDEV360: Secure Software Engineering

Professor David Gabello

22 November 2023

University of Maryland Global Campus

Threat modeling is an extremely useful process in software security, as it allows developers to predict, to the extent possible, the potential threats that an application may face. As defined by Rorak (2023), “It is essentially a risk management process that helps software developers anticipate potential vulnerabilities and security weaknesses in their applications.” (Understanding Threat Modeling, para.1). Just as any kind of modeling, there is a guide developers follow to correctly model potential threats. It consists of three distinct steps: Decomposing the application, Determining and Ranking Threats, and, finally, Determining Countermeasures and Mitigation (Conklin, n.d.). This report will describe and demonstrate the threat modeling process for an Automated Teller Machine (ATM).

**Step 1: Decomposing the Application**

The first step of the threat modeling process involves describing the structure and function of an application, and looking for external dependencies, entry and exit points, assets, as well as trust levels. It also encompasses creating Data Flow Diagrams (DFDs) for developers to understand what they are working with (Conklin, n.d.).

An ATM is a complex device designed to help people with financial operations. In terms of physical appearance, it typically consists of:

1. A printer
2. A screen
3. A keypad
4. A card reader
5. A card dispenser

(Bennett, 2023)

An ATM only has one user, the person performing the financial operation. It also has an operator, responsible for guiding the user through it. In abuse cases, it may also have a hacker, who manipulates it with malicious intentions. An ATM allows the user to take the following set of actions:

1. Deposit costs.
2. Withdraw costs in cash.
3. View balance and other account details.
4. Transfer costs between accounts.

(Bennett, 2023)

**Document Owner:** Mark Kardash

**Participants:** Mark Kardash

**Reviewer:** Professor David Gabello

**External Dependencies:**

Conklin (n.d.) describes external dependencies as “items external to the code of the application that may pose a threat to the application.” (External Dependencies, para.1). In simple terms, they are things that an application depends on, and even though they are not in the code, and not controlled by developers, the organization running the application usually still has control over them (Conklin, n.d.). In the table below, several external dependencies for an ATM are listed.

|  |  |
| --- | --- |
| Dependency ID: | Dependency Description: |
| 1 | The ATM is connected to the bank that operates it. Specifically, it is connected to a network, or multiple networks, responsible for transactions between the ATM and the bank (Snow, 2016). |
| 2 | When a card is inserted into, or pressed against, the card reader of an ATM, the data is read and transferred to the host server/processor of the bank (Agarwal, 2021). The task of the host processor is to keep the ATM connected with all the different networks of its parent bank (Bowen, n.d.). |
| 3 | A major external dependency for an ATM is the operator/attacker, who has control of the machine from a remote location and can manipulate it in any way they want (Snow, 2016). |
| 4 | The fourth dependency is any protection measures the network of ATMs may have (ex. Antivirus, Firewall). |
| 5 | Another dependency is any devices that can be used by attackers to interfere with the normal workflow of an ATM and steal user data. This could be either a skimmer, an appliance used to read data off a card (Iven, 2022), a black box, which allows to control cash trays (Snow, 2016), or any other such tool. |

When it comes to software security, external dependencies must be well-monitored and documented. Precisely because they are external, damage inflicted on them or through them may be very difficult to repair.

**Entry Points:**

Entry points are “the interfaces through which potential attackers can interact with the application or supply it with data.” (Conklin, n.d.). In essence, they are the attacker’s “gates” to their target. An ATM has several entry points, outlined in the table below.

|  |  |  |  |
| --- | --- | --- | --- |
| Entry Point ID: | Entry Point Name: | Entry Point Description: | Entry Point Trust Levels: |
| 1 | Card Reader | The card reader reads the data from a user’s card, and sends it to the host processor (Agarwal, 2021) to be processed in the networks (Bowen, n.d.). | 1. Card holder with correct account credentials. 2. Card holder with incorrect account credentials. 3. Operator. |
| 2 | Keypad | The keypad is used by the card holder to enter credentials, particularly their PIN, to access their account (Agarwal, 2021). | 1. Card holder with correct PIN. 2. Card holder with incorrect PIN. 3. Operator. |
| 3 | Operator | The operator counts as an entry point because they can physically access the ATM. | 1. Operator. |
| 4 | Bank | The bank can be the perfect entry point for an attacker, as it gives them a chance to hit their target “from within”. This usually happens when the attackers have an insider working at the bank, such as in a remote attack on multiple machines (Snow, 2016). | 1. Bank employee. 2. Bank customer. 3. Network/ATM repairman. |

Properly guarding and securing an application’s entry points during development is crucial for its integrity, as it will ensure early detection of vulnerabilities, and seal known attack gateways.

**Exit Points:**

Just as entry points are ways for the hacker to “get in”, exit points are where information leaks out of when an attack occurs. They are necessary for any attack involving information disclosure. Exit point and entry point issues are often related, and an improperly secured exit point can lead to major disclosure of data (Conklin, n.d.). An ATM does not have many exit points, unless one considers the personal network that the attacker connects the machine to as one.

**Assets:**

An application’s assets are the things that interest an attacker in it. These can be either physical, such as a list of personal information from a client, or abstract, like the reputation of the company an attack will affect (Conklin, n.d.). Since ATMs deal with money and financial operations, they have a few lucrative assets that make them a popular target. The assets are listed as follows.

|  |  |  |  |
| --- | --- | --- | --- |
| Asset ID: | Asset Name: | Asset Description: | Asset Trust Levels: |
| 1 | Cash in Card Holder Account/on card. | The first, and obvious asset that an attacker would go for are the physical costs that a card holder has in their bank account, or on their credit/debit card, which is accessible through the ATM. | 1. Card holder with valid credentials. 2. Operator. 3. Bank. |
| 2 | Card Holder Credentials | The credentials of a card holder, such their PIN, are also a lucrative target. If an attacker finds a way to read them, they can use them for malicious purposes. | 1. Card holder with valid credentials. 2. Operator. 3. Bank. |
| 3 | Information in Card Holder’s Bank Account | Attackers may also be interested in the personal information that a card holder has in their bank account, such as their name, date of birth, Social Security Number, available balance, etc. These may frequently be used for identity theft. | 1. Card holder with valid credentials. 2. Bank. |
| 4 | Damaging reputation of the bank/Personal conflict. | Some attackers may want to target a bank for personal reasons, rather than financial gain. One such example could be a former employee taking revenge after being fired, a spy sent by a rival bank, or an attacker disguised as a card holder wanting to damage the bank’s reputation for some other reason. This would make it an abstract asset. | 1. Former bank employee. 2. Spy from rival bank. 3. Customer with valid credentials. 4. Customer with invalid credentials. |
| 5 | Decreasing public trust in ATMs. | Another abstract asset is the desire to discredit the ATM machine as a tool. This could be done by a customer who had bad experiences with the machines, or by someone who has created an alternate device they believe to be better in handling finances, and are trying to draw the public’s attention towards it in such a way. | 1. Card Holder. 2. Operator. 3. Bank employee. |

Assets are essentially the reason and motivation behind cyberattacks. Keeping them safe, just like entry points, is imperative for the security of both software, customers, and company. This is especially true for devices like the ATM, which handle monetary transactions, and therefore may be more lucrative to attackers than an average webpage or program.

**Trust Levels:**

Conklin (n.d.) defines trust levels as “the access rights that the application will grant to external entities.” (Trust Levels, para.1). Therefore, they define who can access which part of an application. They are determined for both assets and entry points, as it is essential to know who is permitted to enter the system, or handle important information, to prevent unauthorized operations. The trust levels of an ATM are distributed in various ways between the people and entities using it.

|  |  |  |
| --- | --- | --- |
| Trust Level ID: | Trust Level Name: | Trust Level Description: |
| 1 | Card Holder (User) with Valid Credentials | A card holder who has the correct credentials (ex. Valid PIN) to access their bank account through the ATM. They have the right to withdraw cash, deposit money, make a transfer (Bennett, 2023), and view their account details. |
| 2 | Card Holder (User) with Invalid Credentials | A card holder who does not have the right credentials to access their account, but is still able to activate the machine. |
| 3 | ATM Operator | The operator of the ATM is responsible for guiding the card holder through any of the operations they want to perform. |
| 4 | ATM Maintenance/Repair Technician | The technician has access to certain parts of the ATM to maintain or repair it. |
| 5 | Attacker | Though not necessarily a trust level, an attacker can still gain access to an ATM in various ways. |

A clear, detailed definition of trust levels is a necessary step towards a secure system. They help determine the amount of privileges that each actor in a system’s life is entitled to, and avoid giving one of them too many. In the case of an ATM, this could help prevent anything from financial machinations to identity theft.

**Step 2: Determining Threats to the Application**

As previously mentioned, an ATM’s purpose of handling financial transactions makes it a very desirable target for attackers. Therefore, the threats that can come its way are many, and can be of diverse nature. This section will determine the most common threats for an ATM and explain in detail the risk that each of them poses.

The ATM is vulnerable to various threats as soon as it is activated by the customer. To access their account, a user is typically prompted for the PIN to their card, or their account number. The risks involved with this are both physical and digital. There are many devices, such as skimmers and shimmers, that an attacker could insert into the ATM to read either the PIN or card information. There are also “pinhole cameras”, that could be put there to give the criminals an idea of the machine’s inner workings. In some cases, a false keypad could be installed, so that the PIN could be read by hackers as the user enters it (Iven, 2022). But even without a physical device to help them, attackers can still attempt to retrieve a PIN using a dictionary or brute-force attack (Kurita et al, 2012).

Broken authentication attacks are another major issue for ATM’s. This is when an attacker can access the PIN, account number, and any other passwords or credentials, to successfully enter the system as someone else (Gupta, n.d.). It can occur for several reasons, including the hacker reading data through the physical devices mentioned before. Broken authentication can give the criminals access to customer finances and records, sometimes even leading to identity theft.

A session management attack is when a hacker takes over a customer’s session, and manipulates its outcome (SecureCoding, n.d.). For an ATM, an example of this is when an attacker controls an ATM’s cash trays after inserting a black box, controlling the machine’s cash trays, and manipulating the customer’s actions (Iven, 2022).

The use of an ATM involves handling a lot of sensitive data, as the card holder enters, and receives, sensitive pieces of information. This makes the machine vulnerable to sensitive information exposure. As the name suggests, it is when information gets exposed to entities that should not have access to it. A card holder may enter a PIN and account number to access their account and receive a display of data such as their name, date of birth, balance, and other details. Should this information fall into the wrong hands, consequences may range from loss of finances and decrease of customer trust, to customer identity theft and legal issues for the bank.

**Step 3: Suggesting Countermeasures**

To keep an Automated Teller Machine secure, developers must take an individual approach to each threat. Iven suggests that the first thing to consider is the location of the ATM itself. The presence of good lighting, security cameras, and being in a low-crime rate area can significantly reduce the chances of the ATM becoming a target (2022). Securing the ATM with bolts and closing off access to the ATM case will guard against attacks that use specialized devices, while regular updates of the software will guard against malware and data breaches (Iven, 2022).

As for its virtual components, to protect the machine from broken authentication, password guessing, dictionary attacks, and session management attacks, several valuable actions can be taken. First, the system could ensure safety by giving the user a limited number of attempts to access their account. After the number has been reached, the account should be locked for a specified amount of time (Mitre.org, 2023). A clear, but not too detailed error message could also be displayed if the length of the PIN or account number is incorrect. This helps ensure that the user is notified of their mistake, but does not receive information that could compromise the safety of the ATM.

Finally, to avoid exposure of sensitive data, developers should simply limit the requests of the ATM to information that is necessary for the actions the card holder wants to take. For example, the user should only be asked for their PIN and account number, not their date of birth, when accessing their bank account. To ensure that unintended actions are not taken, the system could also display a confirmation prompt, and an option to cancel, before continuing with a transaction.

When creating any program or application, its developers have a huge responsibility for everything its proper functionality, to its integrity, to the safety of its users. But for devices like the ATM, which handle sensitive personal information and finances, this responsibility increases even more. Spotting and documenting potential threats early in the design process makes their analysis and mitigation easier, which is why threat modeling is so beneficial for software development.

References

Agarwal, T. (2021, October 18). *Automated Teller machine : Block diagram, types & its working*. ElProCus - Electronic Projects for Engineering Students. <https://www.elprocus.com/automated-teller-machine-types-working-advantages/>

Bennett, K. (2023, November 16). *Automated Teller machine (ATM): What it is and how to use one*. Bankrate. <https://www.bankrate.com/banking/what-is-an-atm/>

Bowen, J. (n.d.). *How ATMs work*. HowStuffWorks. <https://money.howstuffworks.com/personal-finance/banking/atm2.htm>

*Brute force: Password guessing, sub-technique T1110.001 - Enterprise | MITRE ATT&CK®*. (2023). MITRE ATT&CK®. <https://attack.mitre.org/techniques/T1110/001/>

Conklin, L. (n.d.). *Threat modeling process*. OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation. <https://owasp.org/www-community/Threat_Modeling_Process>

Gupta, D. (n.d.). *What is broken authentication and how to prevent it?* LoginRadius: Start With Identity, Go Anywhere. Retrieved November 22, 2023, from <https://www.loginradius.com/blog/identity/what-is-broken-authentication/>

Iven, A. (2022, March 8). *Protect your ATM machine with these 5 ATM security tips*. ATM Depot. <https://atmdepot.com/atm-security-tips/>

Kurita, S., Komoriya, K., & Uda, R. (2012). Privacy protection on transfer system of automated Teller machine from brute force attack. *2012 26th International Conference on Advanced Information Networking and Applications Workshops*. <https://doi.org/10.1109/waina.2012.148>

Rorak, T. (2023, September 12). *The Role of Threat Modeling in Software Development: A Cybersecurity Perspective*. DZone. <https://dzone.com/articles/the-role-of-threat-modeling-in-software-developmen#:~:text=Threat%20modeling%20is%20a%20systematic%20approach%20to%20identifying,potential%20vulnerabilities%20and%20security%20weaknesses%20in%20their%20applications>

Snow, J. (2016, September 29). *4 ways to hack an ATM — video*. Kaspersky Cyber Security Solutions for Home and Business | Kaspersky. Retrieved November 22, 2023, from <https://www.kaspersky.com/blog/4-ways-to-hack-atm/13126/>