Context Aware Authentication Using IoT Services

I. Introduction

The purpose of context aware authentication is to protect against unauthorized access to a system in the event that an attacker gets ahold of a user's credentials. Regardless if the credentials were obtained through social engineering or compromising the system at a technical level, context aware authentication will make it extremely difficult for an impersonator to log into the system by checking the "context" of the surrounding Internet of Things (IoT) devices and determining if the intended user is the person attempting to log into the system at a given location.

Currently, this is only practical for highly secure and localized systems where the user has to login on-premises because the sensor actuator network promised by IoT has not been fully realized. However, once IoT networks are common place and ubiquitous computing becomes a reality, location based access control could potentially become viable for a variety of everyday applications such as webmail, credit card transactions, and accessing property. The authentication backend of the proposed context aware authentication system put forward in this document will be referred to by its working name, PinPoint.

In the current state of network and systems security, system designers overwhelmingly rely on passwords in order to authenticate users and prevent unauthorized access to their products. While there have been some advancements in this approach, namely the widespread adoption of multi-factor authentication, relying on passwords for authentication leaves a system open to security breaches through a variety of channels because anyone with the password is assumed to be the correct user. Whether the password is leaked through social engineering or security vulnerabilities in the system itself the end result is the same, once credentials are compromised an attacker has the same access to a system as the user they are impersonating. Attacks involving compromised credentials could potentially be thwarted if the system was able to verify that the user requesting access was indeed who they claim to be.

For some time now, biometrics have been available that would allow users to positively identify themselves, but these are both expensive and difficult to set up for the average person. Therefore, it would seem that users need a simple, transparent, and secure way to log into various systems that can not only prevent unauthorized access, but positively verify a user's identity. Furthermore, any proposed authentication scheme must be easy for the average user as well as cost effective or the adoption of such a method will be in doubt. The Internet of Things (IoT) presents a unique opportunity to rethink and importove traditional cyber security by using the sensor/actuator networks that are becoming increasingly available.

In order to keep the initial implementation streamlined and achievable in the allotted time frame, we will consider a positive response from n/2 sources as a positive confirmation of the user's presence and allow access with a password at that time. This will also be true for any secondary users that the primary user grants access to the system under their account. The major difference being that the temporary user will receive a one-time passcode and their access can be revoked by the original user. Furthermore, the initial implementation will only support devices that are connected over a LAN (or VLAN/VPC while using AWS) and operating on the same subnet as the authentication server. The implementation will also account for users that have forgotten their password, and use the same location-aware IoT confirmation to either reject or accept a password reset attempt.

Users will be able to securely log into a variety of devices and systems with minimal input since their identity will be confirmed by their presence in the location-aware IoT network. A location-aware IoT network or Location-based access control (LBAC) improves security and makes login easier on the user by not having the user login to a system using traditional means. In addition, by having LBAC, systems would be able to re-configure security functions and features based on the user's physical location. LBAC improves security by restricting access to user accounts if the user is not present at the physical location of the terminal. For instance, if Alice is in Cincinnati, there is probably no valid reason for Bob or anyone else to attempt to access her account from outside of the Cincinnati area (or the building for that matter).

However, there are always scenarios where a user might need access to the data contained in someone else's account, so the system will account for those one off situations by allowing the owner to delegate access to their files and instances. This is an improvement over current security practices that would most likely involve

the sharing of a password to facilitate account access for the secondary user. These types of security exceptions will need to be fine tuned in order to minimize both the security vulnerabilities and inconvenience to the users.

II. Security

A. General Security Concerns

In any authentication system security is a major concern as authentication is usually one of the major attack vectors that are explored when attackers are attempting to compromise a system. The need for well thought out security measures is only increased in the case of context aware authentication systems because the process for authenticating a user becomes distributed. Now, instead of trying to guess a password or crack a weak password hash, attackers may attempt to impersonate devices or use a man in the middle attack to make it appear that the user is in fact present for login attempt. Furthermore, once context aware authentication systems become widely distributed, a perception that the extra security provided by the system is enough to prevent attacks could potentially cause users to become more careless in choosing strong passwords.

B. Context Aware Security

The idea of context aware security to gather the "context" of the current login attempt from a variety of devices. These devices could include almost any type of sensor including card scanners, thermometers, motion sensors, and microphones. Using the context provided by these devices, PinPoint would then attempt to determine whether or not the person attempting to log into the system is indeed the person they are claiming to be.

C. Secure Communication

Given The distributed nature of context aware authentication systems, providing secure communication between the various components is essential to prevent a number of common attacks such as man in the middle, device spoofing (pretending to be a valid sensor) and packet injection (inserting fake context into a data stream). Most of these security concerns can be mitigated using well known technologies and authentication methods such as TLS to encrypt data in flight as well as whitelisting devices using fingerprints or IP addresses.

D. Pinpoint Security

PinPoint, which is the central component of the centralized authentication system has a number of administrative security concerns that need to be actively managed in order to prevent unauthorized user access to the applications that it would serve. For instance, the ability to define what context constitutes a positive identification of the user must be tightly controlled in order to prevent an attacker from weakening the requirements for a positive identification. An attack reducing or eliminating the context requirements combined with users' reliance on the system as a safety net (resulting in weaker passwords), could result in highly effective brute force attacks against a login terminal.

III. System Architecture

There are a variety of requirements that must be inherently satisfied due to the nature of authentication systems as well as the unique nature of context aware authentication. By considering these requirements early in the design process and building the system to take these considerations into account, the chance that context aware authentication can become a viable authentication system are greatly increased.

A. Design Patterns

In order to remain as general as possible, PinPoint will utilize an adapter pattern in order to communicate with various IoT devices and api gateways in order to gather user status. For instance, a smart lock and home speaker such as Amazon's Alexa, would have the same method call such as device.get_status(), but the get_status() call will determine the device type and call the appropriate adapter in order to gather the context of the device and determine if the login attempt is valid. Furthermore, all communication classes will be decoupled from the logical implementation of the context aware authentication this will allow for the addition of communication protocols with very little change to the existing code base.The need to support and swap out communication protocols will be essential when dealing with IoT devices that have no standardized communication protocol.

B. Speed

Due to the critical nature of authentication systems, Pinpoint will utilize a number of strategies to remain highly responsive to requests for user verification. One such strategy will be to gather context from devices and store the context in a way which can be quickly retrieved. For example, when a key card is scanned, the card information as well as a timestamp will be stored in the database with a time to live (TTL). Once the TTL has elapsed and the event can no longer be used as context for a login, the record will be deleted from

the database in order to free up room and ensure the speed of lookups remains constant. This assumes that PinPoint will not be used as the source of record for keeping track of events like building access and user interactions with various devices. If PinPoint is expected to perform record keeping duties, a secondary database such as Mongo DB or Dynamo DB could be used to store currently valid context and perform rapid lookups while the main relational database could serve as the source of record for various user interactions.

C. Redundancy

Since authentication systems are generally mission critical and require constant up time, redundancy must be considered during the initial stages of system design. PinPoint lends itself to being implemented in a redundant fashion due to the highly distributed nature of context aware authentication. For instance, the login application programming interfaces (apis), database(s), and context gathering endpoints/tasks could operate on different nodes which could be scaled independently of one another. This means that multiple nodes can be deployed to handle each part of the system, so if any one node goes down the system will still operate. Furthermore, PinPoint will need to consider the possibility of IoT nodes being offline and be able to automatically respond to those situations. One example is when a sensor goes offline either due to hardware failure or a network outage. Given the preceding situation a fallback behavior must be defined so that system continuity is not affected. In that situation, a different sensor or set of sensors could be used to determine the status of the user or the system could start issuing blanket approvals or denials, assuming the username and password entered are correct. Another possibility is that a back second factor comes into play such as SMS codes or phone calls. The course of action depends on the needs of the organization and the capabilities of the system implementing the authentication request.

D. Login Sequence

The diagram below shows the sequence of a user login in three stages. First, the user submits credentials to a login terminal for a given application. This login terminal could be for any application and would only communicate with the PinPoint system in order verify the user's identity and authorization, in much the same way SAML authentication works today. After the user submits their credentials, the application would submit the credentials to PinPoint which would then gather context information from the available IoT s. Based on the context gathered and the defined requirements for a positive identification, Pinpoint will then return an assertion to the application in order to communicate whether the login is approved or denied as well as any user specific data such as permissions levels (in the event of a successful login).

E. Use Cases

There are a wide variety of use cases in which context aware authentication can be employed to increase security. Some have already been mentioned, such as highly secure localized systems that would be deployed by financial and defense companies. PinPoint could also be used to change the habits of organizations that commonly share passwords between users which reduces accountability in the system. By confirming that the user which is trying to access the system is currently in or around the given terminal, it would eliminate or greatly reduce thepractice of password sharing. In fact, PinPoint could be leveraged make cross account more secure by explicitly allowing users to delegate temporary access to their account by issuing one time passwords and confirming that the user receiving the password is confirmed in the context of the system before allowing login.