IoT Security Project Requirements

Sean Caster, Vincenzo Piscitello, Adam Barton, Ryan Howerton, and Terri Hewitt

Senior Design

Fall 2018

Abstract

The goal of this project will be to implement a secured, decentralized meshnet with low minimum resource requirements to enable the connection of a wide variety of low-end devices which lack a general purpose operating system. Our objective will be to investigate the amount of security offered and resources required by various cryptosystems in order to provide organizations interested in securing their low-end device networks with a quantified estimate of the costs and benefits. In addition, we hope to accomplish graceful degradation on our network, meaning that the elimination or temporary unavailability of a given node will have minimal negative impact on the inter-connectivity of other nodes. This will encourage the securing of many resource limited devices which play a critical role in meeting a wide variety of human needs.

October 30, 2018

Contents

1 Introduction

1.1 System purpose

As it stands, a wide variety of mission critical tasks for many utility grids (water, power, etc.) are accomplished by low power embedded devices. The mass quantity of devices required to fulfill the operational needs of these networks has created a systemic problem of sacrificing security to reduce the cost of power. The purpose of our system, Brypt, will aim to provide a solution for architects of these networks to have the best of both worlds. In minimizing the cost of power while maximizing the security of ad-hoc mesh networks will demonstrate numerous benefits.

In addition to offering an implementation of what our investigations suggest to be an excellent general purpose meshnet cryptosystem, we will generate useful data about the costs of resource asymmetric cryptosystems that can be applied to the design of more special purpose meshnets by organizations looking to secure their particular resource distribution optimally. These findings will be published in a final report alongside the implemented network to help in the potential design of such projects.

1.2 System scope

The ultimate goal of this project will be produce the implementation and verification of the secure networking paradigm succinctly named Brypt. Brypt combines the main cornerstones of the software, bridging and encrypting. The implementation will provide a central server, binaries for node endpoints, and a security protocol working on top of the session layer. In completing the work on this project our team aims to provide a novel solution to security in interconnected networks of embedded and general purpose devices. Some of the benefits Brypt will provide include the increased viability of secure cryptographic primitives in power constrained environments across a number of different communication technologies.

1.3 System overview

1.3.1 System context

The primary context of our system will be provided through a cloud based application served through our central server. Within our web application three primary interfaces will be accessible by the user. The first interface will be the base information page for Brypt; this page will contain information about the system, it's requirements, and downloads. The second interface will be the node management screen; these pages will provide provide node network authorization. The final user interface element of our system is the dashboard page which will contain information pertaining to the status of the user's connected clusters and aggregation of the nodal data. The only requirement of the user will be access to a browser capable device. Outside of our application interface, users will need to interact with our embedded devices and/or binaries for their system of choice.

1.3.2 System functions

Our system capabilities will be provided through our central server, networking implementation, and security protocol. The complete system will need to operate over several communication standards, variable weather conditions, and battery constrained environments. An active portal to a full internet connection will be required for or mesh network to make and maintain connections to their respective clusters.

1.3.3 User characteristics

There is a single user class within our system; this permissions class will act as the manager and maintainer of their organization's clusters. There can be multiple users with access to an association's clusters, although there may be one owner. An individual wanting to use our system and application should be familiar with technology, but may come from all different walks of life. We can expect users to have varying levels education and disability, so accessibility should be built into the user interface.

1.4 Definitions

- Control Server: The central server for a network.
- Coordinator: A node on a user's network with the added responsibility of routing the aggregate messages of the cluster and/or acting as a gateway for devices without direct internet access.
- Node: An endpoint in the network.

2 References

3 Specific Requirements

3.1 Functional Requirements

Brypt will require a number of different functional requirements in the operation of our system. The central routing authority will operate on a remote server which will perform verification and authorization of the network nodes. The mode of authentication will use an aggregate or sequential encryption puzzle to verify the potential network node while also providing some level of DDOS protection. The central server interface will request the nodes UUID as input and listen for the individual node to make contact. When a node has been authorized for connection to the network, it will be provided routing information to a coordinator node for its specific communication technology. Given that the node network will be developed based on a multihop mesh network, the new node will perform a series of acknowledgements to be configured into the topology. During this configuration process the node will generate a transmission "line" table to be able to communicate with as many peers it has access too. Utilizing the information of it's cluster the node will be able to perform it's base operation and listen for failed peers. Our system's security protocol will require the ability to encrypt and decrypt messages using the networks alternating cipher key. While our system's primary security protocol will be based on the FAAS algorithms we will need to update it while testing other, less power efficient solutions.

3.2 Usability Requirements

The system should have an interactive, hosted web application. The application should support the latest version of at least one of the following browsers: Chrome, Firefox, or Safari. Authenticated users should have access to the application and use it to monitor the state of the network. The application must be capable of adding new nodes to a specific network cluster. The system should also tolerate invalid data input by the user during account creation or node registration.

At least 90% of users should be able to set up an account and log in to the application without assistance. When logged in, 90% of users should be able to find and navigate to any of the primary pages.

3.3 Performance Requirements

As a sub-minimum requirement, nodes must be able to communicate with authenticity and privacy ensured. Any compromise of either of these characteristics must only prove feasible for a dedicated, physically active adversary capable of compromising a field device's memory or computationally exhausting its potential keyspace. Any failure of our protocol to route and forward messages that would've made it through on an unsecured network can be considered a failure to meet a pre-requirement of our project.

Beyond not breaking things, the point of our project is to improve the resource usage required for privacy and authentication beyond the obvious cryptosystems already available and commonly known. So, as a basic metric, we intend for any cryptosystem we implement to consume fewer resources than the same network configuration running unoptimized RSA encryption/authentication and Diffie Hellman Key Agreement. Between each implementation of a new cryptosystem, we'll recollect the resource consumption data of the same configuration running this vanilla protocol. This will allow us to ensure any changes in the topography of the network haven't skewed data on our alternate cryptosystems.

3.4 System Interface

There must be an implemented interface that allows for creation of user accounts. The implementation of the user account will give the users options to manage their network. This interface must allow for users to add, view, and delete devices associated with their account and network. New devices will be added through this system interface. Part of this interface should be a "dashboard" allowing for users to view the status of their networked devices.

3.5 System Modes and States

Each networked device will be in a connected, disconnected, or connecting/authenticating state in terms of the network. Additionally, to provide graceful degradation on the network, devices will have a shutdown state to notify the network of its intentions. The central server will handle requests and host a web-application concurrently.

3.6 Physical Characteristics

This network will consist of devices using multiple communication types. Devices will be networked in a pseudo-mesh topology within the same communication type. Each one-type-network will be connected with other one-type-networks through a base-station and network coordinators to provide inter-communication. Examples of communication types are LoRa, WiFi (802.11), and BLE. The base-station will have higher computational resources than many of the networked-devices. Networked devices will have varying computational resources as would be implemented in the real world. The base-station will be equipped with extra boards to allow it to communicate with devices using other communication types that it may not support.

Given the variable performance of ranged wireless connections, we expect there will be times when the physical communication of devices fails through no fault of the protocols we develop; however, it's important to us to ensure that the way we use wireless communications in our protocol doesn't negatively influence the functional range of our devices. For this reason, if we encounter any persistent problems connecting nodes within their expected operational range, we intend to re-install the unsecured base implementation of our network and compare performance. By doing this, we can ensure that the problems we experience aren't caused by any design flaws we've made.

3.7 Environmental Conditions

The devices on the network will be able to exist under a variety of climates, depending on the roles they are expected to fulfill. The standard device will be battery powered, but the initial tests will also have an attached solar panel to charge it under direct sunlight. Because of this expected variation in climate and conditions, nodes will be expected to survive normal wear and tear that can be expected in their area, and still connect and communicate with the rest of the network.

3.8 System Security

Key exchange A new node must be able to authenticate a member of the network given some server fingerprint (could be public RSA key, BLAKE pre-image challenge for server, etc.) It must then be able to derive symmetric keys to be used for bootstrapping future symmetric-key authentication and/or privacy.

Authentication A new node must be able to authenticate its messages to a server in any cryptosystem we implement with a given key or key pair. In systems which perform aggregate authentication, a fallback, non-aggregate, authentication technique should be available. This will help (A) determine which in a group of messages is malicious/inauthentic given failed verification and (B) avoid high-cost aggregate verifications when better-equipped base stations are unavailable.

Privacy Encryption technique should not allow for compromise of privacy, even to a Byzantine adversary. This means, not only should our system use AES-CTR correctly (i.e. increment nonces and pad correctly), it should also be resilient to things such as state-induced Key Reinstallation attacks and Man-in-the-Middle attacks (this will also depend upon our Key exchange and authentication requirements being met). Cryptosystem as a whole should maintain privacy, as opposed to only encryption functions used. This means we'll use Encrypt-then-Authenticate to prevent Authentication techniques from leaking data over multiple key uses.

3.8.1 Responding to Threats

DOS attacks:
System will degrade gracefully in event of Denial of Service (malicious, or natural) meaning that if a node loses its normal bridge to a coordinator node, it will broadcast to search for other nodes able to serve as a new bridge. This program state should be under careful scrutiny, since adaptive response to possibly malicious activity opens a large attack surface to MitM attacks

MitM (Man in the Middle) attacks:
System should be able to use a single (possibly unique) fingerprint for each node in order to prevent MitM during Key Agreement. Once Key Agreement has been accomplished, system should continue to authenticate communication so as to prevent any control message from altering its behavior without a signature.

Injection Attacks - SQL:
System will sanitize input on the server side to avoid any database tampering; whether it come from a client, or a malicious interceptor of a query.

Rogue Node Attacks:
Cryptosystem must resist imitation of legitimate nodes by avoiding reuse of keystreams (i.e. any key, nonce pair in an AES-CTR encryption must be used only once).

3.9 Information Management

Information about the nodes, users, and network clusters should be stored in a database accessible to the web application. Any personally identifiable information stored in the database for each user should be kept to the bare minimum required to identify and contact that user.

Not all rows in each table will have values associated with each field. Information stored about the separate network clusters may vary from one cluster to the other. Different network groups may have fields unique to a specific group. The database must be capable of handling some diversity in the information stored for objects in the same table.

3.10 Policies and regulations

As outlined above, the network will be communicating on various radio frequencies. We will be staying away from those frequencies that are highly regulated, such as the ISM (industrial, scientific, and medical) bands. Radio frequencies have been divided between industries by various organizations around the world, such as the U.S. Federal Communications Commission and the International Telecommunication Union. These organizations place rules on, require licensing for the use of, and carefully coordinate specific radio frequencies, so that there is no interference between signals on these bands.

We will be employing technologies that communicate on frequencies outside and between regulated or restricted ranges (such as Bluetooth and Wi-Fi).

4 Verification

4.1 Functional Requirements

Each aspect of the project will be tested individually and as a system. The central routing authority will be tested to ensure that it can properly verify and authorize devices on the network. Our networking implementation will be tested to ensure proper topology is handled with the changing state as well as the transmission of messages. The security protocol tests will ensure key rotation, encryption, and decryption is working properly for production environments. Testing along the way as well as system tests will verify that this functionality works properly. Unit and random testing will be used to validate functionality of software built for this application. Code/branch coverage can be used as metrics for us to evaluate our tests used to validate our software.

4.2 Usability Requirements

On completion of the web application, a brief 1-2 day period of system testing/user acceptance testing should be performed to ensure the web portion of the system works as expected with less than a 5% rate of failure.

The usability of the web application will be evaluated by a select group of beta testers outside of the development team. With minimal to no assistance from the team, beta testers should be able to figure out how to create an account, log in, check the status of the network, and find where to add a new node to the network. A minimum of two of the beta testers should be non-Computer Science majors. Any functionality issues reported during testing must be resolved if deemed necessary by either the development team or the client.

4.3 Performance Requirements

The cost of the cryptosystem should be measured by sending out a request to each node in the network and measuring the overall battery usage of the node when receiving and responding to these requests. A measurement of the computational work performed or required of each node should also be measured as part of the testing.

4.4 System Interface

See usability requirements. Usability will be determined by the system interface.

4.5 System Modes and States

Verification of system modes and states will be implied through the functional network. Tests for adding, connecting, and removing devices to stress the functionality of authentication, connection, and graceful degradation will be done to ensure that system modes and states are working as intended.

4.6 Physical Characteristics

Each device will be tested by authenticating, connecting, operating on the network, and removing from the network. Standard operation of device lifecycles will be stressed to ensure that operation of the network with the devices works as it should. Devices of differing communication types (LoRa, BLE, WiFi) will all be tested in the same way to verify that all communication types succeed in connecting.

Each physical medium will have its operational range tested once the unsecured network is implemented. Though we'll periodically reexamine these ranges when problems are encountered (most likely due to things like changes in weather), we suspect the functional range will remain largely the same regardless of the sequence of 1's and 0's it's used to transmit. Only if we continually experience problems on a physical connection will we update our effectively expected functional range.

Though it'd hopefully go without saying, we'll continually test the base functionality of our network by continuing to move traffic across it for multiple days for each of the implemented cryptosystems.

4.7 Environmental Conditions

We will test our network under variable weather conditions (as they present themselves), and compare it to control data, which will be gathered in as bare of a space as possible.

4.8 System Security

As a base level verification that our protocols are working, we'll simply ensure that a node given normal conditions is capable of initiating the key exchange phase with the server, authenticating itself during that exchange, and finally sending and receiving encrypted messages before ensuring that the decryption process produces the original encrypted messages.

As a more advanced verification, our group will perform various holistic verifications of system security against different active attacks, such as partial DoS, MitM attempts, and attempts to clone a node.

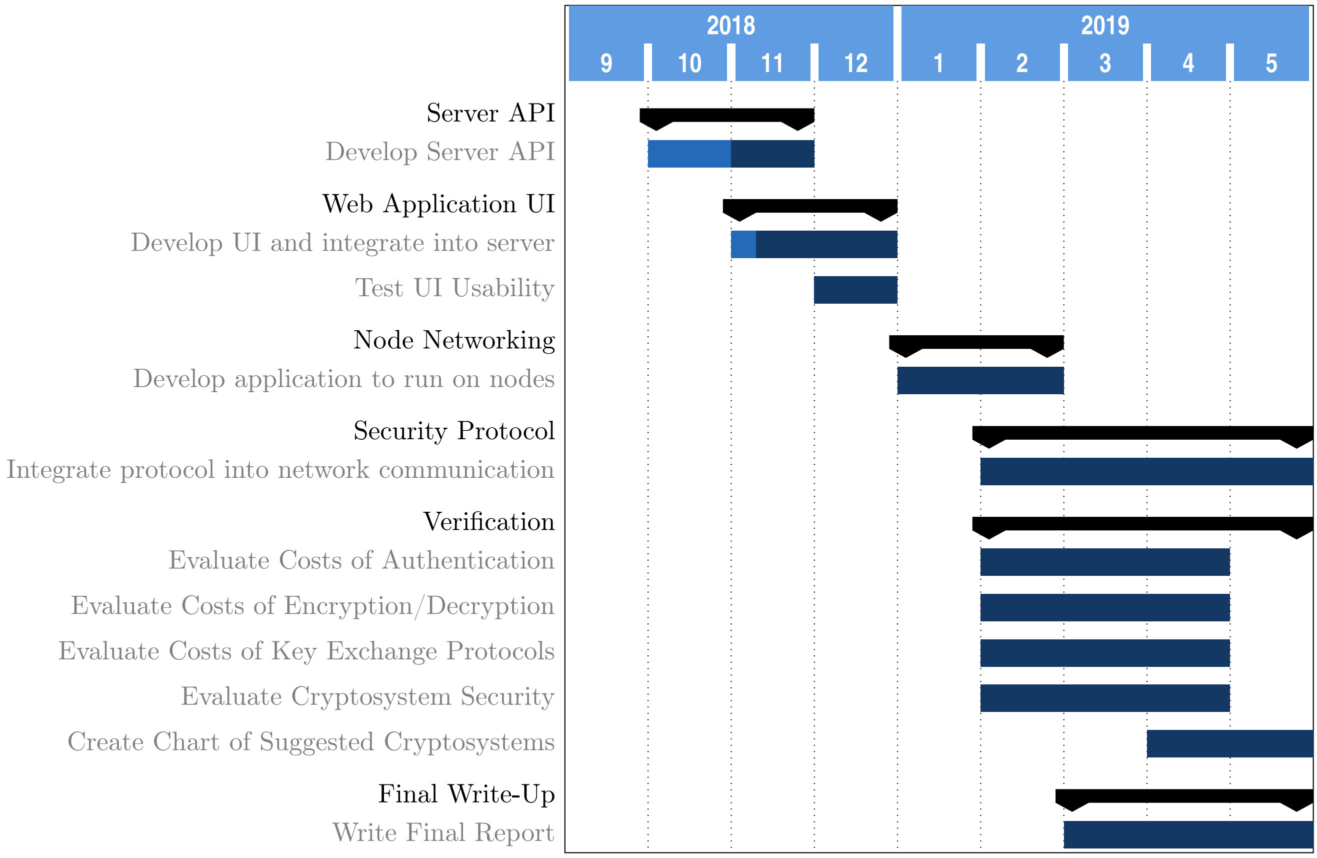
4.9 Information Management

The database(s) will be tested to ensure that information is correctly stored in each table and can be retrieved from each table by the server. Tests should be created to ensure that data cannot be modified during the process of storing it in the database or requesting it from the database. Any URLs and fields which a user could modify that will be processed by the server and sent to the database will be tested to ensure that no disallowed characters or sequences of characters can be present in the input.

4.10 Policies and Regulations

Each node will only be able to communicate on frequencies that are unregulated, as defined by their hardware. We will also test the reception capabilities of the nodes to ensure that there is no interference from outside sources, and that within their natural limits they are able to send and receive to other nodes on the network.

5 Project Timeline



6 Appendices

6.1 Acronyms and Abbreviations

- AES: Advanced Encryption Standard
- API: Application Program Interface
- BLE: Bluetooth Low Energy
- DDoS: Distributed Denial of Service
- DoS: Denial of Service
- FAAS: Fast Authentication with Aggregate Signatures

- MitM: Man in the Middle
- SQL: Structured Query Language
- UI: User Interface
- URL: Universal Resource Locator
- UUID: Universally Unique Identifier