University of Colorado
Interdisciplinary Telecom Program
TLEN 5700

Project Definition Document (PDD)

Operation SUPERFREQ
SDR Automation

Approvals

<table>
<thead>
<tr>
<th>Name</th>
<th>Affiliation</th>
<th>Approved</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kenneth Baker</td>
<td>CU Boulder ITP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(tentative)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Course Coordinator</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kevin Gifford</td>
<td>CU Boulder ITP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Project Customers

Name: Kenneth Baker
Address: 1111 Engineering Drive
        ECOT 318, 530 UCB
Phone: (303) 492-3493
Email: Ken.Baker@Colorado.edu

Name: Joe McManus
Address: 1111 Engineering Drive
        ECOT 318, 530 UCB
Phone: (720) 316-8172
Email: Joe.Mcmanus@Colorado.edu

Team Members

Name: Kade Cooper
Phone: (303) 931-8054
Email: Kade.Cooper@Colorado.edu

Name: Jing Guo
Phone: (303) 956-6856
Email: Jing.Guo@Colorado.edu

Name: Pieter Bork
Phone: (303) 350-8635
Email: Pieter.Bork@Colorado.edu

Name: Carlos Lawrence
Phone: (720) 771-2894
Email: Carlos.Lawrence@Colorado.edu

Name: Jinpeng Miao
Phone: (720) 352-0618
Email: Jinpeng.Miao@Colorado.edu

10/16/2017
1.0 Problem or Need

A Software Defined Radio (SDR) is a dynamic piece of hardware that uses a general purpose CPU and software to interpret and decode digital signals on a range of frequencies. The use of SDRs is not trivial and the learning curve along with the time overhead required to retrieve useful data is a deterrent to users regardless of their skill level. Even a basic FM radio application requires the user to boot to a live image, configure drivers, or install specialized operating systems, requiring hours of troubleshooting and configuring. Current existing SDR packages do not offer a comprehensive solution and most usable features are scattered over a variety of platforms. While modern hardware has made strides towards reducing the size and price of SDRs, current software in SDR space is underdeveloped and is not simple to deploy or use. SUPERFREQ seeks to address these issues by allowing everyday users, developers, or security professionals access to a single platform which produces actionable information using a SDR.

2.0 Previous Work

The use of SDRs is a relatively new concept. The idea started in the 1980’s, but was not fully realized until the 90’s. Many see Joseph Mitola as the founding father of the field, starting with publication of his papers *Software Radios: Survey, Critical Evaluation and Future Directions (1993)* [1] and *The Software Radio Architecture (1995)* [2]. Concurrent to Mitola’s work, the Department of Defense developed the SPEAKeasy project, designed to bring all of the different radios used onto one device [3]. Later in the same decade, the concept was picked up and developed by the newly created SDR Forum, who has continued progress to this day [4]. In 2001 (when the SDR architecture was first patented [12]) Eric Blossom created the GNU Radio Companion to help make common SDRs more accessible to dedicated programmers which was followed in 2004 by the first commercially approved SDR [11], the Anywave, used by cellphone carriers. Mitola continues to drive the field forward [9] and in the past few years, many more SDRs have become available allowing anyone with an interest to acquire the necessary hardware.

Every step of the way, SDR technology has moved towards ease of use and this project continues to push that boundary. The first big step in the field towards ease of access for SDRs was the release of GNURadio [5] which moved developers from pure hardware programming to a simple user interface. In addition to project development software, many people have developed individual projects like BLE Dump [6] (packet capture using a SDR for Bluetooth), BTLE Decoder [7] (packet capture using RTL-SDR for NRF24 and BTLE), and the RFSec Tool Kit [8] (a collection of RF tools). Furthermore, progress is being made to simplify and collect the underlying protocols in a reusable library for SDR development [10]. While these other projects contribute to the utility of SDRs, they are still hosted on version control servers and require the user to configure the underlying systems to enable their use.
3.0 Specific Objectives

**Level 1 (Primary Objectives):** Create an easily distributable package or application that allows for collection and display of wireless SSID, encryption, and signal information on 2.4GHz and 5GHz frequency bands along with collection of Bluetooth information on the 2.4GHz frequency band (collection will be done from an antenna and its base converter unit). The package should be compatible with different distributions on Linux (Windows and macOS are out of scope on this objective due to high learning costs) and be able to setup/maintain its own database saved to a user's hard drive.

**Level 2 (Secondary Objectives):** Software package includes a command line interface and web GUI tools to allow for user interaction and manipulation of data received from the Antenna. All tools are user friendly and do not require a high level of technical knowledge. In addition, the application will decode additional frequency bands including but not limited to Zigbee, LTE, or GSM. This stage will achieve full functionality of hardware identification through obtaining broadcasted identity information such as MAC addresses or hardware model.
4.0 Function Requirements

**SUPER_FREQ** is a package which drives the decoding and manipulation of data packets on a Linux machine obtained from networks it resides in or near.

**Figure 2: Concept of Operations (CONOPS)**

5.0 Critical Project Elements

*Technical:*

**CPE 1.1** Knowledge and creation of a software package that can be deployed on multi-computing platforms. The application can be run on numerous Linux distributions without special preparation using a common scripting language.
CPE 1.2 **Develop interpreting and extracting methods.** The application includes built-in methods to get information from radio frequencies obtained by the antenna.

CPE 1.3 **Develop command line and Web GUI tools for end user manipulation.** Users can use either a graphical user interface (GUI) or command line interface (CLI) to interact with the software.

CPE 1.4 **Deployment of standard security practices (e.g. OWASP) for tools and the software package itself.** To protect users information and create a secure environment, tools will be installed in the software that prevent or handle possible malicious actions.

**Logistical:**

The SUPERFREQ team must:

CPE 2.1 **Acquire software defined radios and dedicated Command Station computer alongside equipment that can output 2.4 GHz, 5GHz, Bluetooth and radio frequencies.** This will be used to test the application and hardware, allowing testing of different Linux distributions and input frequencies.

CPE 2.2 **Have extensive knowledge of scripting languages and Unix/Windows based operating systems.** Software development proficiency and critical thinking skills are be required to configure the hardware and create user friendly interfaces.

**Financial:**

*SUPERFREQ is looking to secure minimum funds of $1000 from the ITP department.*

**6.0 Team Skills and Interests**

**Carlos Lawrence:** Carlos is a Network Security student with a background in networking and computer science. He has experience in web development, user centered software development, UNIX systems, networking hardware and software, and leadership. He is interested in exploring the capabilities of SDRs and applying his computer science background in a networking/security project.

**Pieter Bork:** Pieter is a security enthusiast with a strong software development background. From small software startups to larger security firms, he has been actively developing and maintaining software and systems for over five years. With a Computer Science undergraduate degree focused on Networked Devices & Systems, he believes that SDRs are an up-and-coming device that will be extensively used by security professionals. He is most comfortable working in python but has experience in multiple languages.
**Jing Guo:** Jing is in a Co-op at DellEMC for the past two years. She has experience in data protection and copy data management in agile/scrum environment, and programming experience in Java, Python, PHP and MySQL. She is interested in the software and security side of the project.

**Kade Cooper:** Kade interned at Mentor Graphics for the past two summers. There he had various projects ranging from: programming/deploying automated pen testing scripts, creating a node.js web application, and deploying/maintaining an Active Directory network for authentication based testing. He has prior experience in securing both web and embedded based platforms with regards to the following languages: Python, Perl, PHP, UNIX, MySQL, Java, and C/C++. He is interested in the software and radio aspects of the project, with a specific interest in the packaging of the system.

**Jinpeng Miao:** Jinpeng is currently a member of the Public Safety and NDN research group in the Computer Science Department. His interests include networking architecture and network security with experience in information security and network optimization. Additionally, prior to his graduate studies, he spent one-year working in industry as a network engineer.

<table>
<thead>
<tr>
<th>Critical Project Elements</th>
<th>Team member(s) and associated skills/interests</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 - Creation of Software Package</td>
<td>Kade, Jing, Pieter, Carlos, Jinpeng</td>
</tr>
<tr>
<td>1.2 - Antenna Data Extraction</td>
<td>Pieter, Carlos, Jinpeng</td>
</tr>
<tr>
<td>1.3 - CMD &amp; Web Tools</td>
<td>Kade, Jing, Carlos, Jinpeng</td>
</tr>
<tr>
<td>1.4 - Security Considerations</td>
<td>Kade, Jing, Pieter, Carlos, Jinpeng</td>
</tr>
<tr>
<td>2.1 - SDR Equipment</td>
<td>Pieter, Carlos, Jinpeng</td>
</tr>
<tr>
<td>2.2 - Knowledge of UNIX/Scripting</td>
<td>Kade, Jing, Pieter, Carlos, Jinpeng</td>
</tr>
</tbody>
</table>
7.0 Resources

<table>
<thead>
<tr>
<th>Critical Project Elements</th>
<th>Resource/Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 - Creation of Software Package</td>
<td>Joe McManus, App Image Github</td>
</tr>
<tr>
<td>1.2 - Antenna Data Extraction</td>
<td>Kenneth Baker, Michael Ossmann video tutorials</td>
</tr>
<tr>
<td>1.3 - CMD &amp; Web Tools</td>
<td>Joe McManus</td>
</tr>
<tr>
<td>1.4 - Security Considerations</td>
<td>Joe McManus</td>
</tr>
<tr>
<td>2.1 - SDR Equipment</td>
<td>Command station software, Antenna &amp; decoder module, and wireless equipment</td>
</tr>
<tr>
<td>2.2 - Knowledge of UNIX/Scripting</td>
<td>Joe McManus, Kevin Gifford</td>
</tr>
</tbody>
</table>

Annex

Design Drivers:

1. All software and hardware must include a setup time of five minutes or less in order to enhance end user usability experience.
2. Software must have an easy to use graphical interface in order for the end user to quickly and effectively manipulate network data obtained from antenna.
3. All software must be able to comply with OWASP standards.

Requirements & Constraints:

1. The system shall include the ability to export data from the data collected in variety of standard data formats through automation (e.g. JSON, CSV, etc.).
2. The system shall compile a report to alert users of any new/existing hardware or personal equipment that is detected within a network.
3. The system shall be written in programming languages which are easy to read and maintain.
4. The system shall fit within a single executable program.
5. The system shall cater to both technical and non-technical users through command prompts and web interfaces.
6. The system shall demonstrate the ability to extract hardware information from devices emitting a wide range of frequencies (e.g. 2.4GHz, 5GHz, Bluetooth, etc.).
7. The system shall include persistent storage in the form of an internal database.
8. The system shall only be used with open source libraries and models to ensure interoperability between a wide range of hardware and operating systems.
9. The system shall not collect end user information in any way.
10. The system shall detect possible threats in the network through traffic analysis.
11. The project shall be developed using only open sourced software.
12. The project shall be completed under the GNU General Public License.

8.0 References

[8] RFSEC-ToolKit, cn0xroot, github.com/cn0xroot/RFSec-ToolKit