Simulating the Deployment of Battery-Sensing Intrusion Protection Systems

Theresa M. Nelson, Timothy K. Buennemeyer, Randy C. Marchany, and Joseph G. Tront
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061
{tnelson, timb, marchany, jgtront}@vt.edu

Abstract

This paper extends Battery-Sensing Intrusion Protection System (B-SIPS) research by utilizing network simulations for deployment validation and optimization. The primary simulation goal is to ensure that B-SIPS does not negatively affect external applications in the network, as any drastic throughput degradation would severely lower the probability of successful B-SIPS deployments. The research goal is accomplished goal by modeling the Virginia Tech wireless-cum-wired network and simulating various network sizes, external network loads, and B-SIPS application transmission settings. This research demonstrates that under reasonable network loads the B-SIPS application had little to no effect on the throughput of external applications. Additionally, the 1 second default transmission rate for B-SIPS was determined to cause the least application degradation for external applications and ensured B-SIPS reports were successfully transmitted in a saturated network environment. Next, the detection capabilities of B-SIPS are examined by conducting Bluetooth, Wi-Fi, and blended attacks against mobile devices. The ability of B-SIPS to detect multi-vector attacks provides application users with the ability to conserve battery charge life and retain device service significantly longer than devices undergoing similar attacks and not utilizing B-SIPS. The attacks used in this portion of the research should be applied to future network simulations of B-SIPS. These simulations will quantify network throughput and device battery usage in large scale network deployments that are, and are not, using B-SIPS.

1. Introduction

The primary challenges in developing defensive applications such as intrusion detection systems (IDSs) for small, wireless computers are limited processing capability, memory, and battery resources. Traditionally, network and host-based IDSs employ rules to detect known malicious activity. Anomaly detection systems (ADSs) use statistical methods to establish a system profile and then trigger alerts when that normal profile is violated. This research initiative extends a battery-based detection system that employs mobile devices as sensors that use an instantaneous current-based threshold algorithm to indicate anomalous activity.

An indicator that a rogue process is being run on a device without the knowledge of the user is an unexplained increase in the instantaneous current drawn from a device's battery. This could indicate anomalous activity such as a worm spread, virus infection, network probing, flooding, or denial of service (DoS) attack. All of these malicious activities can cause the battery current to rise such that a well-designed system could detect the illicit activity. The *Battery-Sensing Intrusion Protection System* (B-SIPS) detection capability provides security administrators (SAs) in a network environment with a complementary IDS tool. This nontraditional method detects anomalous battery exhaustion, IEEE 802.15.1 (Bluetooth) and IEEE 802.11 (Wi-Fi) attack activity that standard IDSs are incapable of detecting [1].

The research goal is to validate B-SIPS by showing that its deployment will not adversely affect the network throughput of external applications. If external application throughput degraded severely, then users would most likely disable B-SIPS, rendering it an impractical network security solution. B-SIPS was tested in varying network conditions and scenarios in order to gain confidence that B-SIPS would not cause severe network degradation. Additionally, aspects of the B-SIPS application that may increase or decrease throughput degradation of the network must be explored and optimized. Thus, the *application transmission rate*, or the rate the clients running on mobile devices transmit data to the B-SIPS server, was varied in simulations for optimization determination.

The rest of this paper is structured as follows. Section 2 reviews background and related work. Section 3 reviews assumptions and presents the construction of the network simulation in OPNET. Section 4 discusses simulation testing in networks maintaining a reasonable load. Section 5 extends the investigation of B-SIPS effects in networks already experiencing throughput degradation from large network loads. Section 6 presents Bluetooth, Wi-Fi,

978-0-7695-3450-3/09 $25.00 © 2009 IEEE

and blended attacks tested in the research lab, and explains why they should be incorporated into future network simulations of B-SIPS. Lastly, Section 7 provides a succinct conclusion and a direction for future work.

2. Related work

Battery power is an important resource in the wireless domain, especially for small mobile devices. This presents designers with the problem of choosing more security at the expense of greater power usage and potentially less service availability. Establishing secure communication channels through proper authentication could increase service accessibility from a user's perspective, but it may increase the device's computational and transmission requirements, leading to faster battery drain.

The Advanced Power Management (APM) specification is an application programming interface which allowed computer and Basic Input Output System (BIOS) manufacturers to include power management in their BIOS and operating systems (OSs), thus reducing energy consumption [2]. Subsequently, the Advanced Configuration and Power Interface (ACPI) established an industry-standard for interfaces to OS directed power management on laptops, desktops, and servers [3]. The Smart Battery System Implementers Forum offered an open systems communication standard for industry-wide adoption that described data sharing between batteries and the devices they powered [4]. Their Smart Battery Data (SBData) specification was used to monitor rechargeable battery packs and to report information to the System Management Bus (SMBus) [5] [6].

In 1999 Stajano et al. [7] suggested the idea of energy depletion attacks, which they described as *sleep deprivation torture*. An emerging class of attacks, battery exhaustion and denial of sleep attacks represent malicious situations whereby the device's battery has been unknowingly discharged by the malicious attacker, and thus the user is deprived access to information [8]. These attacks exploit the power management system by inhibiting the device's ability to shift into reduced power states.

Martin et al. [8] subdivided sleep deprivation attacks against laptop computers. *Service-requesting* attacks try to connect to the mobile device repeatedly with power draining service requests. *Benign* attacks attempt to start a power demanding process or component operation to drain the battery. *Malignant* attacks infiltrate the host and alter otherwise typical programs such that greater battery resources than necessary are consumed.

Racic et al. [9] demonstrated successful battery exhaustion attacks that transited commercial cellular phone networks to exploit vulnerabilities in an insecure multimedia messaging service, context retention in the packet data protocol, and the paging channel. These attacks drained the device's battery, rendering it useless in a short period of time by keeping it in a busy state. Most concerning is the fact that the cellular phone user and network administrator were unaware that the attack was ongoing. An attack of this nature will use more device power, and thus demonstrates the potential effectiveness of an integrated battery-sensing IDS [10].

Nash et al. [11] developed a battery constraints-based IDS for laptop computers aimed toward defending the system against various classes of battery exhaustion attacks. They leveraged the laptop's robust computational power to estimate power consumption of the overall system and then adapted this concept on a per-process basis as a method for indicating possible intrusions and rogue applications.

For personal digital assistants (PDAs), Jacoby [12] developed a host-centric *Battery-Based Intrusion Detection* (B-BID) solution. B-BID consists of three distinctive IDS applications. For low power devices, the *Host Intrusion Detection Engine* was a rules-based program tuned to determine battery behavior abnormalities in the busy and idle states using static threshold levels. A complementary *Source Port Intrusion Engine* was employed to capture network packet information during a suspected attack. For robust devices, the *Host Analysis Signature Trace Engin*e was used to capture and correlate spectrum signature patterns using periodogram analysis to determine the dominant frequency and magnitude (x,y) pairs. This system supplied the technological community with a first effort at protecting PDAs with a battery-based IDS solution. Additionally, the system provided a powerful launching point from which new research endeavors could both branch out from and improve upon.

Buennemeyer [1] [10] [13] [14] [15] developed the B-SIPS concept to examine Bluetooth, Wi-Fi and Blended attacks against PDAs and smart phones, and then further developed a net-centric IDS solution that incorporated a dynamic threshold calculation at the client, safe processes list checking, mobile device current profiling at the server as an enhanced means for reducing false positive alerts, and correlation of B-SIPS attacks with the *Snort* IDS. This extension provides not only the B-SIPS client application user with information pertaining to the attack, but also forwards this information to a *Correlation Intrusion Detection Engine* (CIDE) server, where smart battery data from all mobile devices in the network can be

correlated. Network administrators are then able to use this correlated data to draw conclusions pertaining to the state of security in their network. This involvement of not only device users, but network administrators as well, makes B-SIPS a more viable solution for the mobile technological world of today.

The B-SIPS client application was deployed on ten mobile devices and the CIDE server was deployed on a Windows XP machine. Data was collected for attack detection, device battery depletion, and network throughput tests. Lab configurations, however, do not realistically support large scale testing. Network simulations would be required to assert the feasibility of B-SIPS deployment.

Large scale network simulations allow developers to make simplified assumptions regarding the impact of an application on a specified network. Such simulations provide statistics on theoretical networks that would otherwise be impossible to obtain due to a lack of network hardware. To date, there are two network simulation tools that are widely used in research.

The first of these tools is ns-2. Ns-2 is a discrete event simulator targeted at networking research [16]. It is an open source Linux-based simulation tool that uses tcl files and C code to configure simulated, wired and wireless networks. The maintenance of the code repository is largely left to the open source community, who frequently release patches to increase the functionality of the simulator [16].

The second network simulation tool available is OPNET, which is a commercial product [17]. OPNET provides a suite of GUI modules that allow the user to configure networks by specifying the applications running on each network node, the (x,y) coordinates of each network node, the exact router models used and link capacities. Tutorials and product documentation outline usage for available OPNET modules [17].

3. Simulation construction

This research extension creates and analyzes large scale network simulations for determining the likelihood of success in B-SIPS deployment endeavors. To accomplish these simulations, a network simulator was selected, as described in Section 3.1. Next, a simulated representation of a potential deployment network needed to be constructed. As a large technology-driven university, Virginia Tech was determined to be a prime candidate for future B-SIPS deployment. The network topology of the campus is presented in Section 3.2. Simulation tests were devised to thoroughly examine the effects of B-SIPS on external applications, as discussed in Section 3.3.

3.1. Selecting a network simulator

Both ns-2 and OPNET were investigated for their compatibility with the simulation goals of B-SIPS. Ns-2 was found to fair well for low-level network alteration simulations, but it was determined to be infeasible for large scale simulations due to several incompatibilities between released development patches. OPNET, on the other hand, was found to contain all necessary functionality to accurately construct B-SIPS network simulations. Therefore, OPNET was selected as the network simulator used for exploring B-SIPS degradation effects on networks and to determine optimal transmission rate employments.

3.2. Constructing the network topology

The creation of the B-SIPS simulated network topology required the definition of simulation geographical location and conditions, as well as hardware infrastructure, network size, and network load. The Virginia Tech campus was determined to appropriately model a network prime for test deployments of B-SIPS. University campuses around the globe serve as additional networks prime for B-SIPS deployment, as well as large corporate campuses with a desire to monitor and protect the mobile devices used by their employees. In an effort to simulate a network in which network hardware and configuration data was readily available, the Virginia Tech campus was selected for simulation purposes.

Collaboration with Virginia Tech network engineers indicated that bottlenecks in the network occur between individual devices and building access points, as well as between building access points and building core routers. Therefore, the number of academic buildings modeled was less relevant to the study than determining the throughput degradation due to the bottlenecks occurring in individual buildings. In an effort to reduce the time and memory resources required of the machine running the OPNET, the simulations were focused on mobile devices located in two of the academic buildings, Whittemore and Durham Halls. Virginia Tech academic buildings are representative of many corporate high-rise structures and educational campuses because of the high density of IEEE 802.11 B/G Wi-Fi access points and mobile devices that are supported. Whittemore and Durham Halls, in particular, are especially well equipped to handle large quantities of mobile devices due to their close relationship with the Bradley Department of Electrical and Computer Engineering. Many of the faculty and students in department are outfitted with several mobile devices, each of which are pressed to its

limits as system users utilize email, chat services, file transfer applications, and even play peer-to-peer games. The quantity and quality of mobile device use in these academic buildings make them prime candidates for representing stressful conditions in large scale network deployments of B-SIPS.

Network engineers from Virginia Tech provided hardware infrastructure details such as the location of core routers, the number of switches per academic building, and the capacities of network links [18]. In the simulated OPNET scenarios, access points were distributed evenly amongst building floors, as shown in Figure 1 [19]. A random number generator was then used to assign each simulated mobile device to a building and a floor with (x,y) coordinates.

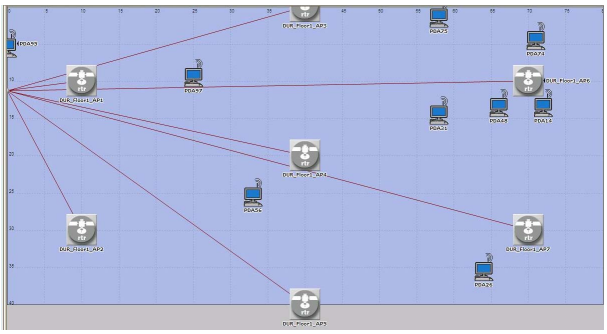

Figure 1. OPNET simulated building floor

The quantity of mobile devices present in an OPNET simulation is defined as the *network size*. The size of a saturated network is determined by multiplying the number of access points by the number of mobile devices permitted to connect to each of those access points. As specified by the Virginia Tech network engineers, there are a total of 60 access points in the simulated buildings, and the suggested number of permitted mobile devices for each access point is 30. Therefore, simulations containing 1800 mobile devices are considered device saturated.

The network load associated with an OPNET simulation is based upon the network size. The B-SIPS application configuration, device hardware parameters, and the number and types of external applications running on each mobile device are incorporated factors in the simulation. The application configuration determines the rate at which smart battery data [4] is sent from mobile devices to the B-SIPS server. A static string of data is transmitted by the mobile devices, regardless of the application configuration setting. This setting, rather, determines the period associated with transmitting this data, as depicted in Figure 2 [19].

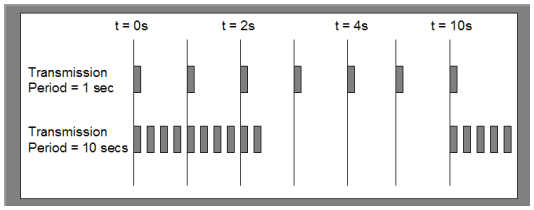

Figure 2. Application transmission period

As shown, two of the optional settings for the application transmission period are one and ten seconds, respectively. If the application transmission period is set to one second, then one application packet is transmitted from the mobile device to the B-SIPS server every second. Alternatively, if the application transmission period is set to ten seconds then ten application packets are transmitted to the server, back to back, once every ten seconds. The application transmission setting is altered during subsequent simulation iterations in order to determine its effect on the network.

3.3. Design of network simulation tests

Finally, external applications running on simulated mobile devices must be considered. In order to thoroughly test the B-SIPS application in a network, tests must be conducted with large and small network sizes, varying application transmission rates, and varying network load. To simulate this circumstance, two distinct test cases were run. In the first scenario, mobile devices running B-SIPS will also run a File Transport Protocol (FTP) client, which is characterized by OPNET as having a *heavy network load*. For this scenario, we will determine the transmission success rate of both the FTP-based and B-SIPS applications. Results are discussed in Section 4.

B-SIPS is tested in networks already experiencing *throughput degradation*, or the reduction of throughput in one application due to interference caused by another application. In this secondary simulation scenario each mobile device is initially running both an FTP client and a Voice over Internet Protocol (VOIP) client, such as Google Talk or Vonage. These two applications were chosen due to a high probability of use by mobile device users and their opposing nature with regard to their network protocols. FTP uses a Transmission Control Protocol (TCP), which provides applications with guaranteed service, thus affording the application lossless data transmission. VOIP, on the other hand, utilizes the User Datagram Protocol (UDP), which is a best effort service and may cause a loss of data when used in device or load saturated networks. Like the FTP client, OPNET characterizes VOIP as having a heavy network load.

Running these two applications in conjunction with one another will cause throughput degradation, and thus fulfills the baseline of this test. Next, the B-SIPS application is started in each simulated mobile device and the throughput degradation is calculated, as shown in Equation 1 [19]. First, the *throughput baseline*, or the throughput an application experiences without external applications running, is calculated. Next, the *test throughput*, or the throughput an application experiences when external applications are running, is determined. Finally, the difference between the throughput baseline and the test throughput is divided by the throughput baseline and multiplied by 100. The percentage calculated represents the throughput degradation due to the external applications added during the testing simulation.

Equation 1. Throughput degradation

$$D = \frac{Throughput\_Baseline - Throughput\_Test}{Throughput\_Baseline} * 100\%$$

For testing purposes, we determined both the throughput degradation experienced by B-SIPS, due to the external FTP and VOIP applications, as well as the throughput degradation experiences by the FTP and VOIP clients due to the B-SIPS application. Results are presented in Sections 4 and 5.

4. Device saturated simulation results

Initial OPNET simulations categorized the effect of B-SIPS on external network applications with respect to the quantity of operating mobile devices. In this testing scenario, the external application is selected to be FTP, and the effect of the B-SIPS application on the network is determined by calculating the percentage of an application's traffic that is received by its intended destination host. The number of mobile devices simulated ranged from 100 to 2000, which intentionally exceeds the determined 1800 mobile device saturation limit. This calculation was performed for each application during each scenario in which both the FTP-based and the B-SIPS application was running on all mobile devices. Characterized results pertaining to these simulations are shown in Figure 3 .

Regardless of the network size, the FTP application was not adversely affected by B-SIPS. The B-SIPS application experienced minimal data loss. The worst scenario simulation lent a B-SIPS reception rate greater than 97%. Additionally, there appeared to be no correlation between the number of devices in the network and the reception rate of FTP or UDP transmissions of B-SIPS.

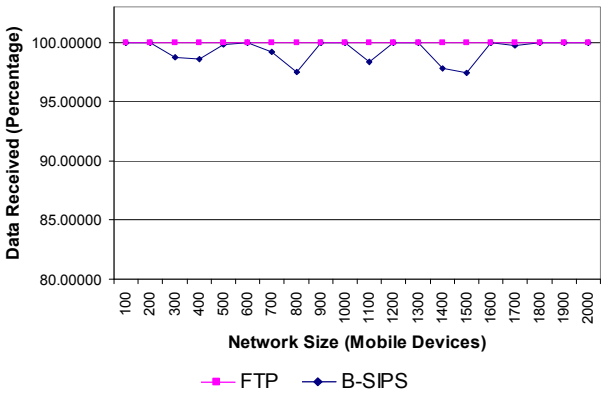

Figure 3. B-SIPS effect on network

The importance of the data from Figure 3 is that, the B-SIPS deployment on networks that are not already experiencing network degradation is not adverse. B-SIPS throughput is high, while the throughput of external applications is even greater. Thus, this data supports and substantiates the feasibility of large scale B-SIPS deployments.

5. Load saturated simulation results

Secondary network simulations were performed to validate the OPNET B-SIPS application model. Additionally, the testing heavily saturated the simulated network, such that the base network is experiencing application throughput degradation prior to the addition of the B-SIPS application. Once the baseline throughput was calculated, the B-SIPS application is added to each client in the network and the degradation due to the B-SIPS application is determined.

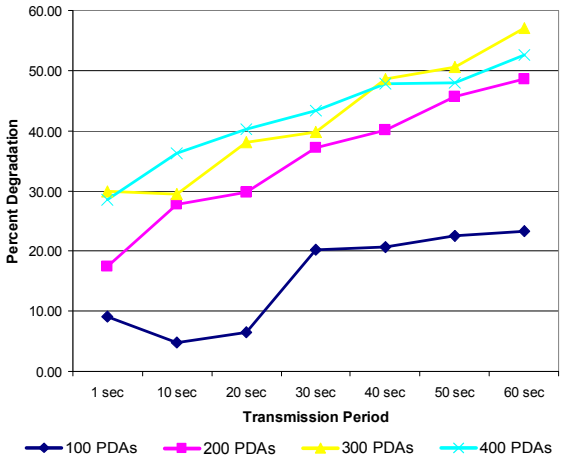

Figure 4. B-SIPS throughput degradation

Each simulated mobile device in the network runs both VOIP and FTP clients. This allows the simulation to both accomplish the baseline degradation and explore the effect of B-SIPS on both UDP and TCP

applications. The effects of a saturated network on B-SIPS data is discussed prior to the effect of B-SIPS on external network applications. The B-SIPS results are portrayed in Figure 4.

The throughput degradation of B-SIPS data increases as the number of devices increases. This behavior can be attributed to the additional transmission interference that is generated by the newly added devices' network traffic. The increase in throughput degradation as the transmission period increased from 1 to 60 seconds can be attributed to the UDP nature of the B-SIPS application packets. Reducing the size of the transmission period produces a greater number of network packets, but affords each packet with a lower interference probability. This phenomenon held true in the OPNET simulation testing of B-SIPS, and provides a validation that B-SIPS encountered the least amount of throughput degradation when the transmission period was set to the default value of 1Hz.

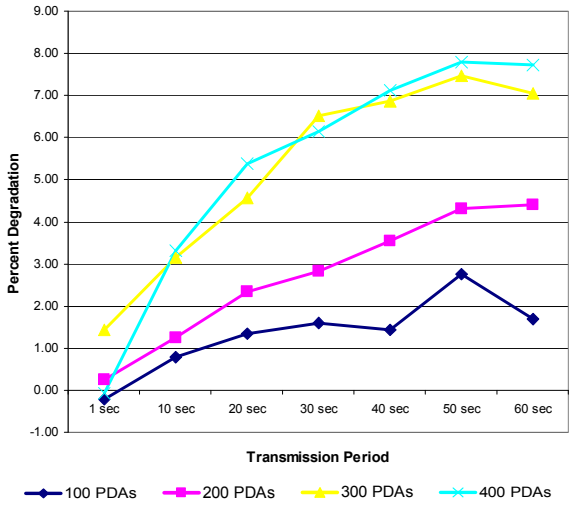

Figure 5. VOIP throughput degradation

The effect of B-SIPS on external applications is determined. Figure 5, shown above, depicts its effects on external UDP applications, such as VOIP. Similar to the B-SIPS application, VOIP experienced an increase in throughput degradation as the network size increased, as well as an increase in degradation as the application transmission period increased. Likewise, this suggests that the minimization of the network size and the use of the default 1 second interval for transmitting B-SIPS application packets will lend the most promising throughput of external UDP applications.

Finally, the effect of B-SIPS was determined for external TCP applications like FTP clients. As shown in Figure 6, the TCP applications do not follow the same patterns as the UDP applications reviewed previously. Data pertaining to the FTP client does not appear to follow trends in regards to network size or application transmission periods.

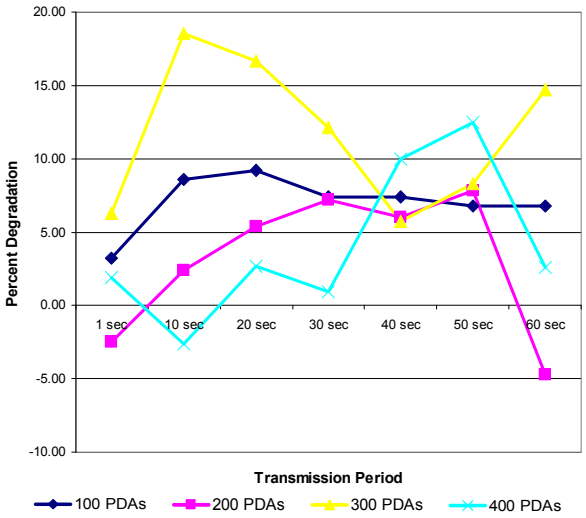

Figure 6. FTP throughput degradation

When comparing Figure 3 and Figure 6, a conjecture can be made that the state of the network prior to the addition of B-SIPS is a determining factor on how adversely the application addition will affect the network throughput. In Figure 3, the simulated network had no network saturation prior to the addition of the B-SIPS application. Thus, the degradation of the FTP application was minimal. In Figure 6, however, the simulated network was already saturated, due to the FTP and VOIP applications on each device competing for network bandwidth. In this case, adding the B-SIPS application had a much more drastic effect on the throughput experienced by the FTP client. This is expected, as sparse resources are being reduced.

The reason for the lack of linear degradation in Figure 6 is unknown, though a conjecture has been made that the irregular data is due to the inclusion of retransmission and control packets in OPNET throughput calculations. Future B-SIPS research endeavors are encouraged to explore this phenomenon in more depth.

6. Attack testing with B-SIPS

The simulations presented in Sections 4 and 5 do not test the ability of B-SIPS to detect and report malicious traffic, but rather, the simulations determine the throughput degradation occurring in non-malicious networks. Additional lab testing has confirmed that B-SIPS successfully detects a number of Bluetooth, Wi-Fi, and blended network attacks. This section

discusses the attacks that were tested and how they can be incorporated into future B-SIPS simulations.

Table 1. Bluetooth attack types

Attack Type	Exploit	Threat Level	Focus
BlueSnarf	Authentication	Medium	Information Theft
BlueBug	Hidden Channel	Medium	Root Control
Helomoto	Authentication	Medium	Hijacking
BlueFish	Authentication	Low	Surveillance
Car Whisperer	Weak Passkey	Low	Eavesdropping
BlueSmack	Buffer Overflow	High	DoS
Bluetooth Stack Smasher (BSS)	Buffer Overflow	Medium – High	DoS
BlueSYN *	Buffer Overflow with SYN Flood	Medium – High	Blended DoS
BlueSYN Calling *	Buffer Overflow with SYN Flood	Medium – High	Blended DoS
PingBlender *	Buffer Overflow with Ping Flood	Medium – High	Blended DoS
BlueSniff	NA	Low	Service Discovery
RedFang	NA	Low	Service Discovery
BlueScanner	NA	Low	Fingerprinting
BTScanner	NA	Low	Fingerprinting
BlueJacking	NA	Low	Messaging

* Denotes original attack crafting.

Until recently, many Bluetooth device users considered their systems to be safe from attack. Table 1 [14] indicates that Bluetooth exploits and emerging attacks are increasing. The ever changing state of attack vectors has opened another avenue for attack signature development, which encompasses the characterization of Bluetooth wireless personal area network (WPAN) attacks. Small mobile computers often have Bluetooth capabilities enabled in discoverable mode out of the shipping box; therefore, devices are vulnerable from the moment they are activated. Moreover, these devices typically lack antivirus software, IDS, and firewall capabilities, so they are unprotected against attack vectors. With new exploits being discovered regularly, a window of opportunity is available for B-SIPS to help protect PDAs and smart phones from attacks.

Bluetooth and other blended attacks will become more prevalent as flaws are discovered and exploited. This will offer greater opportunities to develop battery-based trace signatures, since Bluetooth capable devices tend to be of low powered design. Attacking exposed hosts through unsecured WPANs would allow the attacker direct access to the mobile device and its OS environment, completely bypassing any upstream defensive measures [14]. This observation suggests that mobile devices have an increasing need for hybrid IDS protection such as B-SIPS.

While considering plausible attacks that could disrupt PDA and smart phone operations, three original DoS attacks were crafted in the lab: *BlueSYN*, *BlueSYN Calling DoS* and *PingBlender* [14]. These blended attacks were designed to saturate the test device's multiple communication channels' capabilities to hasten the drain time of its battery resources. BlueSYN is executed by simultaneously attacking the device with a *hping2* SYN flood, affecting the Wi-Fi interface, and a *l2ping BlueSmack* flood, affecting the Bluetooth interface. It demonstrates a blended attack that attempts to saturate multiple communication vectors. The SYN flood propagated through a wired LAN to an access point before finding the target device, while the Bluetooth portion of the attack was launched from a Bluetooth adapter on the notebook computer directly against the targeted device. This previously undocumented attack was named the BlueSYN DoS, and its trace is shown in Figure 7 [14]. This crafted DoS attack is considered to be a medium to high level threat because it can exhaust the target device's resources and severely limit both Bluetooth and Wi-Fi capabilities.

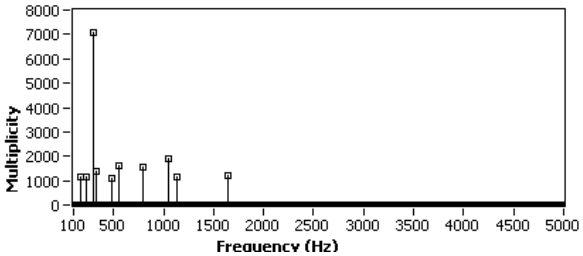

Figure 7. Crafted BlueSYN DoS attack trace signature for Dell Axim X51

For cellular smart phones specifically, extending the above attack and combining it with rapidly dialed phone calls (war dialing) and/or text messages causes the device to react abruptly. This attack requires significant reconnaissance because the attacker needs to know the target device's IP address, MAC address (discovered through fingerprinting), and cellular phone number. This original blended DoS attack was crafted in the lab and was coined the BlueSYN Calling DoS. Although unlikely that this DoS attack will be launched outside the laboratory setting, it is considered to be a medium to high level threat because of its resource exhausting and crippling communication effects on the targeted device.

The PingBlender DoS employs a Ping Flood combined with BlueSmack. The combination forms a blended attack that keeps the mobile device in a higher state of busy to rapidly exhaust its battery resources. Most networks allow ping packets at modest delivery rates, so the crafted PingBlender attack is feasible with modest reconnaissance.

The success of the B-SIPS application in detecting each of the attacks discussed in this section is notable. In order to gain complete confidence of potential B-SIPS customers, however, the users will want to

know not only how B-SIPS performed against Bluetooth and network attacks in a lab, but how B-SIPS would perform in a large scale deployment. Table 2 presents the Bluetooth, Wi-Fi, and blended attacks that were executed in B-SIPS trace testing. These attacks are representative of categories of attacks and exploits that can be conducted against mobile devices. The blended attacks demonstrate multi-vector attacks that exploit concurrent Bluetooth and Wi-Fi propagation. Many of the tested attacks and exploits rapidly deplete smart battery resources, significantly reducing battery charge life, and rendering the mobile device unserviceable until its battery is recharged.

Table 2. Tested attacks

Attacks By Family	Category	Propagation
Bluetooth		
BlueSmack	DoS / Flood	Bluetooth
BlueSnarf	Unauthenticated Data Theft	Bluetooth
Bluetooth Stack Smasher (BSS)	Buffer Overflow	Bluetooth
Car Whisperer	Open Interface Eavesdropping	Bluetooth
PSM/BTScanner	Device Fingerprinting	Bluetooth
RedFang	Service Discovery	Bluetooth
USSPPush	File Injection	Bluetooth
Wi-Fi		
SYN Flood	Flood	Wi-Fi
Ping Flood	Flood	Wi-Fi
Blended		
BlueSYN *	Buffer Overflow and SYN Flood	Concurrent Bluetooth and Wi-Fi
PingBlender *	Buffer Overflow and Ping Flood	Concurrent Bluetooth and Wi-Fi

These attacks were launched from an attack script run on a notebook computer with Bluetooth and Wi-Fi adapters and a Wi-Fi access point.
* Indicates original blended attacks developed during B-SIPS research.

Future OPNET network simulations will be necessary to fully explore B-SIPS capabilities and limitations in large scale networks that are experiencing malicious traffic. Accomplishing this knowledge will require launching Bluetooth, Wi-Fi, and blended attacks on each of the devices in the network. Additionally, simulation results can be verified by comparing the results obtained in the lab with those produced by small scale simulations. Once verification is complete, large scale simulations can be constructed and run. Their results will help predict the proficiency of B-SIPS at suppressing attacks and will also predict the accuracy of the CIDE server under periods of sever network duress.

7. Conclusion

This paper introduces large scale network simulations of the B-SIPS application using the commercial product OPNET. The purpose of this simulation endeavor was to determine throughput degradation rates caused by B-SIPS data transmissions, determine optimal transmission frequencies for various deployment scenarios, and use this data to determine whether or not the deployment of B-SIPS is advisable. The network simulation was designed to reflect the wired-cum-wireless topology structure of the Bradley Department of Electrical and Computer Engineering at Virginia Tech, with mobile devices running B-SIPS being located in both Durham and Whittemore Halls. Small alterations to the topology allow the simulations run in this research to be applied to any other representative educational, research, or corporate campus networks. Simulations indicated that external applications experience minimal to no service degradation unless the network was extremely saturated. In such oversaturated networks, it was determined that the default 1 Hz B-SIPS data transmission period minimized the degradation experienced by both the B-SIPS and external applications. Ultimately, the OPNET simulations provided the necessary confidence to promote the deployment of B-SIPS in large scale network scenarios.

The OPNET simulation scenarios will allow future smart battery researchers to learn about the present state of the mobile device in varying network densities. This knowledge will, in turn, assist system designers to protect the battery from malicious charge depletion and could also help B-SIPS defend running device applications from being altered, corrupted, or eavesdropped upon.

B-SIPS research investigated Bluetooth, Wi-Fi, and blended attacks to gain an appreciation of possible attacks that could confront PDAs and smart phones. The selected devices were connected to an oscilloscope and battery readings were sampled during various attacks. The sample was then converted from the time domain to the frequency domain, and then further filtered to identify unique (x,y) pairs as indicators of the trace signature. These signatures are stored in a database and in the future they will be used for attack correlation with the Snort IDS.

Future work in this area is plausible via an extended exploration of the effects of B-SIPS on TCP-based applications operating in load saturated networks. Additionally, future simulations should incorporate the network attacks presented in Section 6. This will provide potential customers with a more accurate view of how networks will react to attacks whether or not mobile devices are running B-SIPS.

8. References

[1] T. Buennemeyer, F. Munshi, et al., "Battery-sensing intrusion protection for wireless handheld computers using a dynamic threshold calculation algorithm for attack detection," in *40th Annual Hawaii Int'l Conf on System Sciences (HICSS-40)*, IEEE Computer Soc., Waikoloa, HI, 2007.
[2] Microsoft, "Advanced power management v1.2," http://microsoft.com/whdc/archive/amp_12.mspx, 2001.
[3] "Advanced configuration and power interface," http://www.acpi.info, 2005.
[4] "Smart battery system implementers forum," http://www.sbs-forum.org, 2005.
[5] "System management bus," http://smbus.org, 2005.
[6] E. Thompson, "Smart batteries to the rescue," http://www.mcc-us.com/SBSRescue.pdf, 2000.
[7] F. Stajano and R. Anderson, "The resurrecting duckling: security issues for ad-hoc wireless networks," in *7th Int'l Workshop on Security Protocols*, Cambridge, UK, 1999.
[8] T. Martin, M. Hsiao, et al., "Denial-of-service attacks on battery-powered mobile computers," in *2nd Annual IEEE Conf on Pervasive Computing and Communications*, Orlando, FL, 2004.
[9] R. Racic, D. Ma, et al., "Exploiting MMS vulnerabilities to stealthily exhaust mobile phone's battery," in *15th USENIX Security Symposium*, Vancouver, BC, 2006.
[10] T. Buennemeyer, M. Gora, et al., "Battery exhaustion attack detection with small handheld mobile computers," in *IEEE Int'l Conf on Portable Information Devices (Portable '07)*, Orlando, FL, 2007.
[11] D. Nash, T. Martin, et al., "Towards an intrusion detection system for battery exhaustion attacks on mobile computing devices," in *3rd IEEE Int'l Conf on Pervasive Computing and Communications Workshops (PerCom '05),* Kauai Island, *HI*, 2005.
[12] G. Jacoby, R. Marchany, et al., "Using battery constraints within mobile hosts to improve network security," *Security & Privacy Magazine, IEEE,* vol. 4, pp. 40-49, 2006.
[13] T. Buennemeyer, T. Nelson, et al., "Polling the smart battery for efficiency: lifetime optimization in battery-sensing intrusion protection systems," in *IEEE Southeast Conf*, Richmond, VA, 2007.
[14] T. Buennemeyer, T. Nelson, et al., "Battery polling and trace determination for bluetooth attack detection in mobile devices," in *8th Annual IEEE SMC Information Assurance Workshop*, West Point, NY, 2007.
[15] T.Buennemeyer, T. Nelson, et al., "Mobile device profiling and intrusion detection using smart batteries," in *41st Annual Hawaii Int'l Conf on System Sciences (HICSS-41)*, IEEE Computer Soc., Waikoloa, HI, 2008.
[16] K. Fall and K. Varadhan, "The ns manual: the VINT project, a collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC," http://www.isi.edu/nsnam/ns/doc/index.html, 2008.
[17] OPNET Technologies, Inc., "Documentation," http://www.opnet.com, 2008.
[18] C. Gaylord, "Interview with senior network architect at Virginia Tech," 2008.
[19] T. Nelson, "Optimizations of battery-based intrusion protection systems," in *Electrical and Computer Engineering*, MA Thesis, Virginia Tech, 2008.