Effects of Wi-Fi and Bluetooth Battery Exhaustion Attacks on Mobile Devices

Benjamin R. Moyers, John P. Dunning, Randolph C. Marchany, and Joseph G. Tront
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061
{bmoyers, jpvt40, marchany, jgtront}@vt.edu

Abstract

This paper provides insight into the ramifications of battery exhaustion Denial of Service (DoS) attacks on battery-powered mobile devices. Several IEEE 802.11 Wi-Fi, IEEE 802.15.1 Bluetooth, and blended attacks are studied to understand their effects on device battery lifetimes. In the worst case, DoS attacks against mobile devices were found to accelerate battery depletion as much as 18.5%.

Also presented in this work is a hybrid Intrusion Detection System (IDS) designed to thwart this form of malicious activity; Multi-Vector Portable Intrusion Detection System (MVP-IDS). MVP-IDS combines host-based device instantaneous current (IC) monitoring with attack traffic signaturing modules.

1. Introduction

Modern lives are becoming ever more dependent of mobile devices. Cellular phones, Personal Digital Assistants (PDAs), and smart phones, also known as Portable Information Devices (PIDs), keep people in constant contact with friends, family, co-workers, and the World Wide Web. These kinds of devices have a user expected battery charge lifetime. Therefore, power conservation in mobile devices is of paramount concern. If device life can be prolonged, users can be more productive and more satisfied with the use of the device.

2. Related Work

An expectation of prolonged battery life has led to the development of the Smart Battery System (SBS) [1]. SBS is a system used to control, monitor and conserve battery power in mobile devices ranging from PIDs to mobile medical equipment. A smart battery utilizes embedded electronics to store smart battery data (voltage, current, remaining capacity, run-time-to-empty, etc...) and operating parameters, which in turn allows the SBS to predict and optimize battery performance for extended mobile device run-times [1].

Advanced Power Management (APM) [2] and Advanced Configuration and Power Interface (ACPI) [3] have been created for the purpose of standardizing power conservation techniques through the use of industry-common configuration interfaces. APM is a Basic Input Output System (BIOS) - based layered software standard that allows higher-level software to interact with operating systems and device drivers to reduce power consumption without the need of knowing the hardware interfaces [2]. The main idea behind APM is to control power usage of a system based on system activity, meaning if system activity decreases, so does the power to system resources.

ACPI is an industry specification that builds upon the older APM standard to further enhance programming interfaces for power management purposes. The purpose of this specification is to create an industry-wide standard for the configuration of motherboard power management.

Power management can be enhanced and standardized on an industry-wide scale with the creation of the APM and ACPI specifications. This not only simplifies the realm of power management in computer systems, but also improves performance and allows for longer operating lifetimes of those devices using battery-powered hardware.

Even with the SBS, APM, ACPI, and other power conservation techniques, attackers are exploiting mobile devices through DoS attacks targeting rapid battery depletion. These attacks are known as sleep deprivation torture, or battery exhaustion attacks [4] [5]. When mobile devices are inactive, or not in high need of system resources, they enter sleep mode. This allows the device to enter a state of minimal power consumption and process suspension. If an attacker can keep a mobile device in a high rate of power consumption without allowing it to sleep, the device will become inoperable much faster than expected due to insufficient battery resources. It becomes difficult to detect and defend against with this being a novel approach for a DoS attack.

Martin et al. [5] further investigated this approach of attacking mobile devices and introduced

IEEE computer society

978-0-7695-3869-3/10 $26.00

a way to classify different variations of these attacks. Based on their classification system, battery exhaustion attacks could be implemented in three different ways:

- *Service Request Power Attacks:* This category of attack targets battery depletion by attackers making repeated request to victim devices for certain services. These services are usually network-based, aimed at draining the battery through increased Wi-Fi communication on the wireless Network Interface Card (NIC).
- *Benign Power Attacks:* An attack of this nature forces victim devices to repeatedly perform tasks that consume vast amounts of battery power. This form of attack is hidden to the user, but it is not something that is intended to harm the device in any way other than to accelerate the process of battery depletion. Requiring a mobile device to execute hidden Java script is an example of a benign power attack.
- *Malignant Power Attacks:* This type of attack is not only aimed at draining the battery, but also being harmful to the overall operation of the device. Attacks of this form can make changes to the operating system kernel or change application binaries so that more power is drained during execution. These attacks are usually implemented in the form of viruses or Trojan horses and target increasing CPU clock frequency.

Researchers have begun to put forth effort to try to mitigate the effectiveness of battery exhaustion DoS attacks on mobile devices. Nash et al. [6] observed a need for detecting battery exhaustion attacks and produced a viable prototype for laptop computers that could accomplish this on a per process basis. This approach used system performance parameters, such as, CPU load, disk reads/writes, and network transmissions to first estimate correlation coefficients using a multiple linear regression model. By doing so, the coefficients could then be used to model and estimate power usage of the system as a whole. Battery exhaustion can be detected when power expenditures exceeded the estimation for extended time periods using the power estimation model. Another feature of the IDS is the ability to map power consumption on a per process basis. Each process is monitored to allow for the detection of processes consuming large amounts of processor usage. Since processor usage is the largest factor in power consumption, a process aimed at battery exhaustion would have a higher processor usage than that of most other processes on the system.

Jacoby also attempted to solve the problem of battery exhaustion attacks by creating the Battery-Based Intrusion Detection System (B-BID) [7] [8] [9]. This was the first power monitoring anomaly-based IDS intended for securing PIDs. B-BID incorporates three modules to monitor power consumption and to correlate anomalies with network based connections. The *Host Intrusion Detection Engine (HIDE)* module is a rules-based engine that attempts to detect power expenditure abnormalities based on device power consumption characteristics and static thresholds based on PID power states. The *Scan Port Intrusion Engine (SPIE)* module is used to monitor incoming/outgoing network connections, network interface statistics, and routing tables. The *Host Analysis Signature Trace Engine (HASTE)* module attempts to identify attacks based on energy expenditure signatures created using a Fast Fourier Transform (FFT).

Jacoby was able to produce a viable solution for securing PIDs against battery exhaustion attacks with the invention of B-BID. Directly from his research evolved the Battery-Sensing Intrusion Protection System (B-SIPS) [10]. B-SIPS is a client/server-based model used to also detect power anomalies in PIDs developed by Buennemeyer et al. This research extended Jacoby's work by introducing the DTC algorithm [10] which attempted to mitigate false positive intrusion alerts by analyzing power consumption characteristics and recalculating device power expenditure thresholds every second.

3. MVP-IDS Design

The main objective of this research is to hinder outside sources from negatively influencing the usability and lifetime, per battery charge, of PIDs. Since PIDs are dependent on mobile battery sources with limited lifetimes, attacks focused on draining battery life can, in effect, produce a DoS on these devices [4, 5].

MVP-IDS was built directly from B-SIPS. The original B-SIPS design demonstrated that anomalous IC drains, representing attacks against PIDs, could be recognized and reported to a centralized server for forensic analysis. Moreover, in doing so, it also introduced and uncovered grounds for further research in the attempt to validate these IC drains with actual wireless attack traffic.

The methodology and goal behind MVP-IDS is simple: recognize a significant change in IC on a PID and correlate the change with malicious Wi-Fi or Bluetooth traffic. MVP-IDS, shown in Figure 1, was developed to function as a hybrid IDS and is divided into four distinct modules:

1. *B-SIPS Client*: The B-SIPS client [10] is a host-based application that continually monitors PID IC for anomalous behavior in the attempt to detect wireless attacks aimed at draining excessive battery power. The B-SIPS client is used as the triggering mechanism for MVP-IDS.
2. *Snort-Based Wi-Fi Module*: The main objective of this module is to ensure that all Wi-Fi traffic has the ability to be monitored and analyzed for attacks.
3. *BADSS Module*: BADSS was built to monitor and recognize Bluetooth attacks. It does this using the Merlin II Bluetooth protocol analyzer [11] to sniff traffic and the *BADSS Intrusion Detection Engine* (IDE) to match Bluetooth traffic patterns with attack signatures contained in its attack signature database.
4. *Correlation Intrusion Detection Engine [10] (CIDE) Server*: The CIDE server functions as the supervisor for the system, performing attack correlation and developing grounds for administrative action. Once the CIDE server has information regarding the correlation of an IC anomaly with an associated attack signature from Snort or BADSS, it sends administrative responses back to the attacked PID.

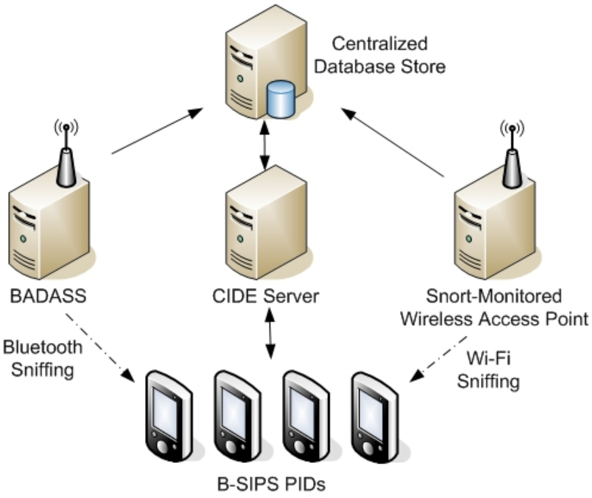

Figure 1. MVP-IDS system overview

4. MVP-IDS Testing and Results

Incrementally, as MVP-IDS and its components were built, tests were run to ensure that the system functioned properly. The main scope of this research focuses on being able to recognize attacks, but to do so, in order to prolong battery life.

4.1. B-SIPS Client Deployment to PIDs

The B-SIPS client was deployed to six Dell Axim X51 PDA's. This was done in order to not only have a set of devices to gather data with, but also to have the ability to compare data within a device set. The Dell Axim X51 PDAs each have the following specifications [10]:

- Microsoft Windows Mobile 5.0
- Intel XScale PXA270 Processor at 520 MHz
- 3.7" 640x480 color TFT VGA display
- Wi-Fi and Bluetooth wireless technologies
- 64 MB SDRAM and 128 MB Flash ROM
- Removable 1100 mAh Li-Ion Primary Battery

4.2. Attack Tools

To properly test MVP-IDS and all of its components, this research took full advantage of the many penetration testing tool kits widely available on the Internet. The main attack suite consisted of a PC running Backtrack 3 [12] configured with 2 USB Bluetooth dongles and an active connection to the subnet created by the Snort-Based Wi-Fi module.

The attack tools used for launching Wi-Fi attacks included *hping3*, *nmap, Nessus3,* and *Unicorn scan*. The Bluetooth attack tools used in this research included: *RedFang, Btscanner, BluePrint, PSM Scan, RFCOMM Scan, BlueBug, BlueSnarf, Btcrack, CarWhisperer, BlueSmack, Nasty vCard, L2CAP Header Overflow, HCIDumpCrash, Bluetooth Stack Smasher, Ping of Death, Tbear, Helomoto, Nokia N70 DoS, Tanya attacks, BlueSpam,* and *Blueper.*

4.3. Data Collection

There needed to be a way to accurately monitor device lifetimes, while not requiring user interaction for the battery exhaustion trials. To do this, a time logging application was developed that appended the current time to a text file at one second intervals. The time logger could then be used to monitor device lifetimes once deployed to the PIDs. When the PID's battery resources were fully depleted, the device would shutdown, thus terminating the time logger application. The previous device lifetime could then be easily obtained by subtracting the first time recorded in the time log from the last time.

4.4. Battery Drain Testing Setup

The main objective of this research was to hinder outside sources from negatively influencing the usability and lifetime, per battery charge, of PIDs.

Battery lifetimes of PIDs under different operating conditions had to be evaluated to determine the effectiveness of the MVP-IDS system. Buennemeyer first explored this area by examining battery lifetimes of Dell Axim X30 PDAs under idle conditions, running the B-SIPS client, and also testing the device under attack from a SYN flood. Next, he examined Dell Axim X51 PDAs to determine the most power conservative B-SIPS client status reporting rate. While these tests provided an initial starting point for examining the effectiveness of the B-SIPS client and PID battery lifetimes, there were still many other operating conditions that were not investigated. Rapid battery depletion due to Bluetooth, Wi-Fi, and blended attacks are the main focus of this research, mainly because these are vectors which are most vulnerable on PIDs. This research further explores these venues and tries to reinforce the idea that the B-SIPS client not only protects PIDs from wireless attacks, but also protects their associated battery lifetimes.

All devices were fully charged and set to the maximum performance state, meaning that the backlight was never dimmed and the processor was at maximum operating speed. The Wi-Fi and Bluetooth radios were also enabled during each trial, though network connections were only established as required to perform an attack. The time logger application was deployed to each PID to record the battery lifetime for each device in each trial.

While a maximum performance testing scenario may not accurately portray the average user's needs, its purpose is to illustrate that battery exhaustion attacks can be significantly detrimental to the battery lifetime of PID's. The devices used in this experimental setup demonstrated the ability to sustain battery lifetimes of days if configured to power conservation modes. However, due to time constraints, all devices were configured to maximum performance states to allow for more trials to be examined in a shorter time interval, thus permitting more of a breadth to the variety of attacks.

4.5. PID Battery Drain - Idle

Each PID was tested under idle conditions to determine a baseline value representing its optimum battery lifetime for comparison in later battery drain trials. Once each device was fully charged and appropriately configured to its maximum performance state, a trial was started. The time logger application was allowed to run for the duration of the battery drain trial so that when a PID was fully discharged, a total battery lifetime could be recorded. This process was repeated for 15 trials, using 6 different Dell Axim X51 PDAs. Two predictions were made regarding the results.

1. Each PID should produce its own consistent data set with very little deviation. The data set for each device should show a normal distribution with actual time trials clustering around the mean battery lifetime for each device set.
2. The battery lifetimes for each device should vary only slightly from device to device. This means that the difference between battery lifetimes sets for each device should not be statistically significant within a 95% confidence interval.

Table 1 shows the data obtained in this test set. To determine the validity of the predictions made before testing, two different statistical methods were used. Prediction 1 was analyzed by graphing each device's set of battery lifetimes on a normal quantile plot. As Figure 2 shows by the diagonal trend lines, each device successfully conforms to a normal distribution. Also, the standard deviation for each set of device trials is very low, all under 2 minutes. For the scope of this research, two minutes added or subtracted to a PID's battery lifetime is not significant. Therefore, the mean value from each set of trials can be used as a representative number for approximating that device's battery lifetime.

Table 1. Idle condition battery lifetimes (Sec.)

	PDA 1	PDA 2	PDA 3	PDA 4	PDA 5	PDA 6
Mean	6246	7343	6218	6743	6749	6208
St. Dev.	31	113	97	116	98	91

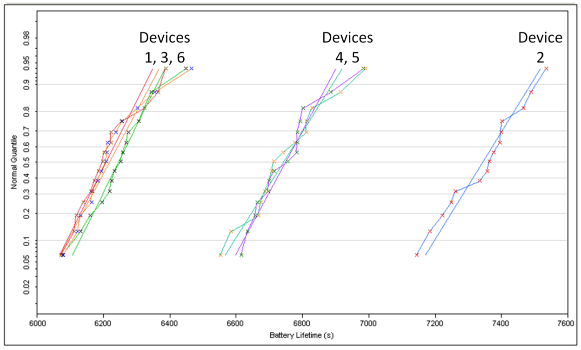

Figure 2. Normal quantile plot of PID battery lifetimes under idle conditions

A One-way Analysis and Student's t-test was performed using a 95% confidence interval in order to assess prediction 2. The results of this test are shown in Figure 3, which represent statistically similar time trial sets as overlapping circles.

Contradictory to prediction 2, two physically identical devices from the same manufacturer will not always produce battery lifetimes that are statistically similar. While this is a surprising conclusion to prediction 2, there is a possible explanation. Buennemeyer noted that through battery drain trending [10], PID batteries continually lose their charge capabilities on successive charge/discharge cycles. A PID that has a larger battery lifetime could simply be indicating that the device is newer than devices it is being compared to.

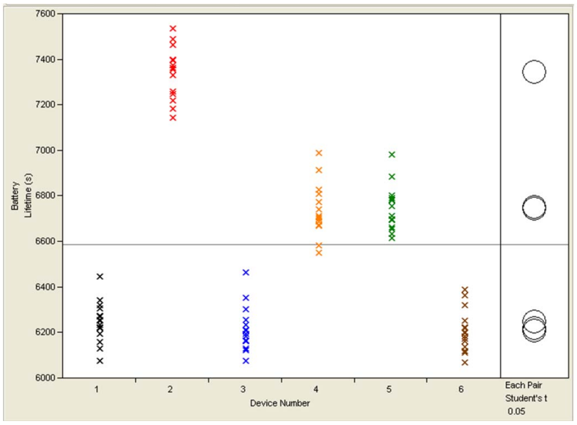

Figure 3. Analysis of PID battery lifetimes under idle conditions

4.6. PID Battery Drain – B-SIPS Client

Previously, Buennemeyer had obtained battery lifetimes for the original B-SIPS client with a 10 second reporting rate. These values could not be used in this research effort, since the lifetimes did not take into account the time logger application or the new code changes that have been made to support the bi-directional communication mechanism of MVP-IDS. Because these modifications could greatly affect battery lifetimes, new testing had to be performed.

PID battery depletion lifetimes under idle conditions were established as a baseline to compare all other successive tests to. With this benchmark in place, the B-SIPS client was then tested for efficiency. The B-SIPS client must not have a significant negative impact on a PID's battery lifetime for it to be successful in the mobile environment. The test setup used for this set of time trials is similar to that used to obtain battery lifetimes of PIDs under idle conditions. All devices were again fully charged, configured into their maximum performance states, and timed using the time logger application. The MVP-IDS version of the B-SIPS client was started and allowed to continually run for the entire duration of the test. The recorded battery lifetimes obtained from the time logger application are shown in Table 2.

Table 2. Battery lifetimes running the B-SIPS client (Sec.)

	PDA 1	PDA 2	PDA 3	PDA 4	PDA 5	PDA 6
Mean	6089	7212	6110	6603	6545	6072
St. Dev.	110	78	105	94	106	86

The results of Table 2 show that PID battery lifetimes are less when running the MVP-IDS version of the B-SIPS client, but only approximately 2.2% less than that of the device alone operating under idle conditions. Table 3 shows the battery lifetime differences between the means of idle PID trials and those running the B-SIPS client. Also noticed in this series of tests was that battery lifetime of PIDs running the MVP-IDS version of the B-SIPS client also conform to a normal distribution, as shown in Figure 4.

Table 3. Battery lifetime comparisons (Sec.)

	PDA 1	PDA 2	PDA 3	PDA 4	PDA 5	PDA 6
Idle	6246	7343	6218	6743	6749	6208
Client	6089	7212	6110	6603	6545	6072
% Diff.	2.51	1.78	1.74	2.08	3.02	2.19

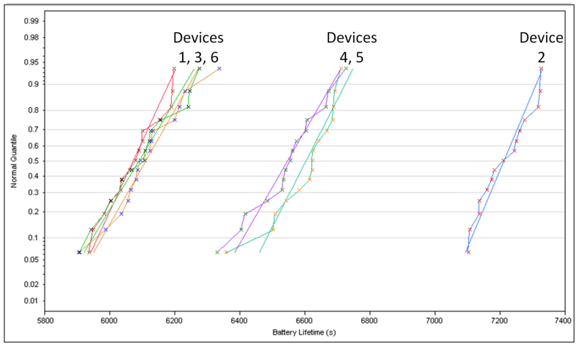

Figure 4. Normal quantile plot of battery lifetimes for PIDs running the B-SIPS client

4.7. PID Battery Drain – Wi-Fi Attacks

Two important details were discovered from the baseline testing of PIDs under idle conditions, as well as test results from running the MVP-IDS version of the B-SIPS client.

1. Not all PIDs tested produced similar battery lifetime means. Therefore, when analyzing results from other tests, time trials could only be compared to baseline values on a per device basis.
2. Since each PID tested produced battery lifetimes that were normally distributed for both previous time trial sets, it was decided that fewer time trials were needed to accurately convey the mean battery lifetime value for other tests. For the

remainder of the battery lifetime testing, 5 trials were used to calculate a mean value approximating a PID's battery lifetime.

The focus of battery drain testing turned to battery exhaustion attacks via the Wi-Fi medium with these two details in mind. All PIDs were initially configured in the same manner as they were for idle baseline testing. The only difference between those tests and the current set is that another computer launched a particular Wi-Fi flooding attack at the idle PID. From the attacks listed in Table 4, only 3 were chosen as suitable to acquire valuable battery lifetime results. These attacks were a ping flood, an ACK flood, and a SYN flood.

The ping flood was chosen as a testable attack because it is distinctive and produced repeatable full battery discharges. Attacks 2-9 in Table 4 closely resemble each other in attack usage. The only difference between these attacks is the bit flags that are set within the TCP packet header. Attacks 2-9 were successful, but only the SYN and ACK floods were chosen for battery lifetime testing. The SYN flood was chosen as an attack trial because it could be used for comparison during the testing of the BlueSYN blended attack. The ACK flood was chosen as a third attack trial to show that changing a bit, or series of bits, in a TCP packet would not greatly affect the battery lifetime of a PID. A Nessus default scan was not applicable to battery lifetime testing because it was not a flooding attack that would possibly lead to rapid battery depletion. Only attacks that had potential for battery exhaustion capabilities were considered for any battery drain tests. It was also discovered that successive nmap and unicorn scans would cause a malfunction in the PID's Wi-Fi NIC during pre-trail testing. Therefore, these attacks were also deemed unsuitable for battery lifetime testing.

Table 4. Wi-Fi attack suitability determination

#	Attack Name	Suitability For Testing
1	**Ping Flood**	**Successful and Repeatable**
2	**ACK Flood**	**Successful and Repeatable**
3	FIN Flood	Replicates ACK Flood
4	PUSH Flood	Replicates ACK Flood
5	RST Flood	Replicates ACK Flood
6	**SYN Flood**	**Successful and Repeatable**
7	URG Flood	Replicates ACK Flood
8	XMAS Flood	Replicates ACK Flood
9	YMAS Flood	Replicates ACK Flood
10	Nessus Default Scan	Not Applicable
11	Nmap Intense Scan	Crashes Wi-Fi NIC of PID
12	Nmap OS Scan	Crashes Wi-Fi NIC of PID
13	Nmap Quick Scan	Crashes Wi-Fi NIC of PID
14	Unicorn Scan	Crashes Wi-Fi NIC of PID

The PID chosen for all attack testing purposes was based on random selection prior to device identification. Also noted during attack testing was that flooding attacks were not able to produce battery lifetime results. The flood option of *hping3* simply crashed the Wi-Fi NIC and caused it to become unresponsive. To combat this obstacle, the 3 Wi-Fi attacks were run at a much slower speed, only 100 packets per second. The three chosen Wi-Fi attacks are described below and the battery lifetime results are shown in Table 5:

- *Ping Flood:* Ping is a troubleshooting network tool used to determine if a host is reachable on a given network. This tool functions by sending an ICMP echo request to the target host in hopes of receiving a return ICMP echo reply. A ping can act as a DoS flooding attack against a desired host at high speeds of deployment. A ping flood attack was launched at PDA 1 using Backtrack 3 and the command: *hping3 --faster <PDA1 IP Address>*.
- *ACK Flood:* An ACK flood was launched at idle PDA 2, much in the same manner as the ping flood was against PDA 1. This type of flooding attack is created by crafting a packet with the ACK bit set in the TCP header. The command used to implement the attack was: *hping3 --ack --faster <PDA2 IP Address>*.
- *SYN Flood:* A SYN Flood is essentially the same attack as an ACK flood, but with the SYN bit set in the TCP packet header. It was launched using the same method as the previous attacks, but was directed at PDA 3 using the command: *hping3 --syn --faster <PDA3 IP Address>*.

All three Wi-Fi attacks drained their target PDA batteries approximately 10% more than during idle conditions. ACK and SYN flood attacks produced similar battery drain results, as originally predicted. The one percent difference in device battery lifetime could be attributed to the attack being directed at different PDAs.

Table 5. Battery Lifetimes of PIDs under Wi-Fi attacks (Sec.)

PDA 1 Battery Lifetimes			
	IDLE	**Client**	**Ping Flood**
Time	6246	6089	5496
Percent	100	97.49	87.99
PDA 2 Battery Lifetimes			
	IDLE	**Client**	**ACK Flood**
Time	7343	7212	6564
Percent	100.00	98.22	89.39
PDA 3 Battery Lifetimes			
	IDLE	**Client**	**SYN Flood**
Time	6218	6110	5476
Percent	100.00	98.26	88.07

4.8. PID Battery Drain – Bluetooth Attacks

Battery drain testing for Bluetooth attacks was designed with a very similar setup to that of the battery drain testing for Wi-Fi attacks. First, the Bluetooth attacks from the BADSS attack signature database were analyzed to distinguish which attacks were most suitable for battery drain testing. The chosen attacks were Ping of Death, BlueSmack, BlueSpam, and Blueper floods. These attacks produced successful and repeatable results when directed at the Dell Axim X51's. Bluetooth attacks that were deemed unfit for battery drain testing were characterized as so because they either crashed the Bluetooth stack on the device or were not flooding attacks that would rapidly deplete PID batteries. Table 6 shows all Bluetooth attacks and there suitability for battery drain testing.

Table 6. Bluetooth attack suitability determination

#	Attack	Outcome
1	RedFang	Not Applicable
2	Btscanner	Not Applicable
3	Tbear	Not Applicable
4	BluePrint	Crashes Bluetooth Stack
5	PSM Scan	Crashes Bluetooth Stack
6	RFCOMM Scan	Crashes Bluetooth Stack
7	BlueBug	Not Applicable
8	BlueSnarf	Not Applicable
9	Btcrack	Not Applicable
10	CarWhisperer	Not Applicable
11	Helomoto	Not Applicable
12	**BlueSmack**	**Successful and Repeatable**
13	Nasty vCard	Not Applicable
14	L2CAP Header Overflow	Not Applicable
15	HCIDumpCrash	Not Applicable
16	Nokia N70 DoS	Not Applicable
17	Bluetooth Stack Smasher	Crashes Bluetooth Stack
18	**Ping of Death**	**Successful and Repeatable**
19	Tanya	Crashes Bluetooth Stack
20	**BlueSpam**	**Successful and Repeatable**
21	**Blueper**	**Successful and Repeatable**

The four chosen attacks from the Bluetooth attack suitability characterization described in Table 6 were tested for their effect on PID battery lifetimes. Descriptions of these attacks are listed below with battery lifetime results shown in Table 7.

- *Ping of Death Flood:* Much like a Wi-Fi ping, Bluetooth also has a tool, *l2ping*, to determine if a host is reachable. Using *l2ping* at a high rate of speed essentially has the same DoS effect on a Bluetooth connection as it does on a network connection. PDA5 was selected as the device to be tested and was attacked using the following command: *l2ping -f <PDA5 Bluetooth Device Address>.*
- *BlueSmack Flood:* This Bluetooth flooding attack is essentially a Ping of Death attack, but is deployed with a much larger data payload, 600 bytes. Using the 600 byte payload size sometimes causes Bluetooth stacks to malfunction on some devices, but was a successful and repeatable attack against the Dell Axim X51 PDAs. The attack was launched at PDA6 by executing the command: *l2ping -s 600 -f <PDA6 Bluetooth Device Address>.*

Table 7. Battery lifetimes of PIDSs under Bluetooth attacks (Sec.)

PDA 1 Battery Lifetimes			
	IDLE	**Client**	**Blueper Flood**
Time	6246	6089	5154
Percent	100.00	97.49	82.52
PDA 5 Battery Lifetimes			
	IDLE	**Client**	**Ping of Death Flood**
Time	6749	6545	5980
Percent	100.00	96.98	88.61
PDA 6 Battery Lifetimes			
	IDLE	**Client**	**BlueSmack Flood**
Time	6208	6012	5687
Percent	100.00	96.84	91.61
PDA 6 Battery Lifetimes			
	IDLE	**Client**	**BlueSpam Flood**
Time	6208	6012	5201
Percent	100.00	96.84	83.78

- *BlueSpam Flood:* BlueSpam, modified by this research to create vCardBlaster, is an attack that identifies Bluetooth-enabled devices in discoverable mode and spams selected targets with repeated vCard messages. This attack is most often used as an annoyance, but can be classified as a DoS flood if the rate at which the sending of the vCard messages is extremely elevated. vCards were sent as fast as possible to simulate a DoS flooding attack with hopes of depleting a PID's battery very rapidly. The attack was launched at PDA1 using the command: *vcblaster -t 1000 -g <PDA1 Bluetooth Device Address>*. The -t option is the number of times to send a vCard and the -g option tells the program to generate a random vCard.
- *Blueper Flood:* Designed especially for this research effort, this attack resembles BlueSpam in nature, but repeatedly floods a device with file transfers instead of vCard messages. The attack was deployed against PDA1 using the command: *blueper -i 1000 -s 1000 -e -t <file name> <PDA1 Bluetooth Device Address>.* The *-i*

1000 -s 1000 in used to specify 1000 iterations of files with size 1000 byte. The *–t <filename>* specifies the file to be sent and the *–e* option adds a counter to the end of the filename so that files have unique names.

Blueper and BlueSpam were surprisingly successful at draining PID battery sources based on Bluetooth's low power consumption design. Also interesting was the difference in battery drain lifetimes of the Ping of Death and BlueSmack floods. Battery lifetimes of these attacks had a difference of 3% by only changing the packet payload size. Overall, it has been shown that Bluetooth can be an effective medium in which to deploy DoS flooding attacks at battery-powered PIDs.

4.9. Battery Drain of PIDs – Blended Attacks

Blended attacks are those that combine two attack mediums, Wi-Fi and Bluetooth, into a single more powerful attack. In most cases, these attacks are designed with the intention of inflicting far quicker damage to a target device than is possible using only a single attack medium. It was hypothesized for this section that blended attacks would have a much larger negative impact on PID battery lifetimes than single vector attacks.

Currently, there are only two known blended attacks: BlueSYN and PingBlender. These attacks were developed during Buennemeyer's research and appeared as an excellent avenue to explore through battery drain testing in this research. It was learned that blended attacks, do in fact, put greater strain on a device than single vector attacks once testing began. Almost all battery drain time trials were not able to complete because of this. Blended attacks caused the Wi-Fi NICs to become unresponsive to any network communications. To combat this and obtain results, battery drain was monitored on a PID until the Wi-Fi NIC became unresponsive. The time the device became unresponsive was recorded and extrapolated to predict a full battery drain. BlueSYN and PingBlender are described below with battery lifetime results shown in Table 8.

- *BlueSYN Flood:* This attack involves attacking a device by simultaneously launching a BlueSmack *l2ping* flood and an *hping3* SYN flood. The commands used to implement the attack against PDA4 were:
  > *l2ping -s 600 -f <PDA4 Bluetooth Device Address>*
  > *hping3 --syn --faster <PDA4 IP Address>*
- *PingBlender Flood:* Much in the same fashion as BlueSYN, this multi-vector attack uses *l2ping* and *hping3* in attempt to exhaust battery resources of target devices. The difference between this attack and BlueSYN, is that this attack is a combination of ping floods from both Wi-Fi and Bluetooth mediums. The commands used to deploy this attack against PDA2 were:
  > *l2ping -f <PDA2 Bluetooth Device Address>*
  > *hping3 --faster <PDA2 IP Address>*

Table 8. Battery lifetimes of PIDs under blended attacks (Sec.)

PDA 2 Battery Lifetimes			
	IDLE	Client	BlueSYN Flood
Time	6743	6603	5498
Percent	100.00	97.92	81.54
PDA 4 Battery Lifetimes			
	IDLE	Client	PingBlender Flood
Time	7343	7212	6192
Percent	100.00	98.22	84.33

Both PingBlender and BlueSYN were very successful battery exhaustion attacks compared to the entire tested attack set. However, the hypothesis for blended attacks in this section was contradicted. BlueSYN was the most effective battery exhaustion attack, but PingBlender didn't show quite the same success to validate the idea that blended attacks are always the most effective battery exhaustion attack method. Blueper and BlueSpam proved to be very malignant attacks, which is surprising due to the results obtained by the other two Bluetooth attacks. A possible answer to this anomaly is that Blueper and BlueSpam were successful not only because of the heavy Bluetooth traffic loads, but also because of the file saving and memory usage consumed on target PIDs.

4.10. PID Battery Drain - Summary

This research has made three significant conclusions from PID battery drain testing. First, Dell Axim X51 PDA batteries drain in a normal distribution fashion, but the drain time across devices is not always statistically similar. Second, it has shown that battery exhaustion attacks should be seen as a significant threat to the field of mobile device security. The battery lifetimes of these devices need to be treated as dominant asset of the device because without a constant power source, mobile devices are crippled. Lastly, the MVP-IDS version of the B-SIPS client has shown to be effective at mitigating flooding attacks, while consuming very little battery lifetime overhead.

Testing has shown that the B-SIPS client application used an excess of approximately 2.2% of a PID's battery lifetime. However, if the B-SIPS client was allowed to run in the background during a

BlueSYN flooding attack, it could mitigate the attack and preserve as much as 16% of a PID's battery lifetime, as compared with an unprotected PID. A summary of these statistics is shown in Figure 5 and a complete summary of attacks tested in this research is shown in Figure 6.

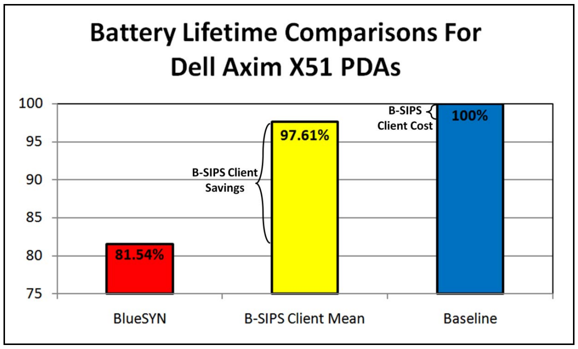

Figure 5. Effects of B-SIPS client and a BlueSYN attack on battery lifetime

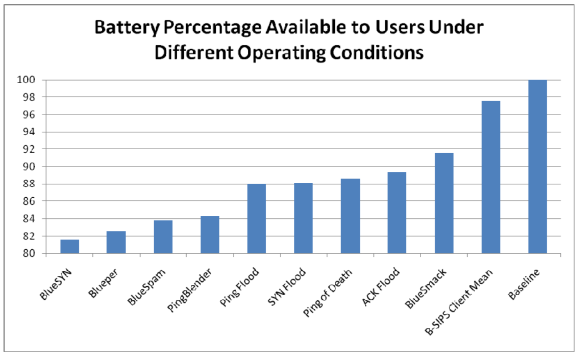

Figure 6. Battery availability to users

The maximum power configuration used in these experimental trials lowers the average battery lifetimes from days to hours. Although it is not certain that the results found will linearly extrapolate to power-conservative performance modes, it is theorized that battery exhaustion attacks will still have significant adverse effects on a PID battery lifetime.

5. Conclusions and Future Work

Extensive testing has shown the possible threat posed by battery exhaustion attacks. This threat can be viewed as a benign nuisance or mission critical failure, depending greatly on the purpose of the victim device. Through testing and analysis, this research shows that extreme battery exhaustion DoS attacks against mobile devices can accelerate battery depletion as much as 18.5% if attacks go undetected during a device's battery lifetime. It was also shown that MVP-IDS can detect and mitigate these attacks, saving up to 16% of a mobile device's battery lifetime.

While this research has provided greater insight into the effect of some battery exhaustion attacks, it still only targeted one specific model of PDA for results. Future tests will examine other mobile devices to determine if the effects of battery exhaustion attacks correlate across device makes and models. Future experiments will also examine battery exhaustion attacks directed at PIDs in power conservative modes.

6. References

[1] Ed Thompson, "Smart Batteries to the Rescue," http://www.mcc-us.com/SBSRescue.pdf, 2000.

[2] Microsoft, "Advanced Power Management v1.2," http://www.microsoft.com/whdc/archive/amp_12.mspx, 2008.

[3] "Advanced Configuration and Power Interface," http://www.acpi.info/, 2009.

[4] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues For Ubiquitous Computing," *Computer,* vol. 35, pp. 22-26, 2002.

[5] T. Martin, M. Hsiao, Ha Dong, and J. Krishnaswami, "Denial-of-service attacks on battery-powered mobile computers," in *Pervasive Computing and Communications (PerCom '04)*, pp. 309-318, 2004.

[6] D. C. Nash, T. L. Martin, D. S. Ha, and M. S. Hsiao, "Towards an intrusion detection system for battery exhaustion attacks on mobile computing devices," in *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on*, pp. 141-145, 2005.

[7] G. A. Jacoby and N. J. Davis, "Battery-based intrusion detection," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, pp. 2250-2255 Vol.4, 2004.

[8] G. A. Jacoby, R. Marchany, and N. J. Iv Davis, "Battery-Based Intrusion Detection: A First Line of Defense," in *Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC*, pp. 272-279, 2004.

[9] G. A. Jacoby, R. Marchany, and N. Davis, "How Mobile Host Batteries Can Improve Network Security," *Security & Privacy, IEEE,* vol. 4, pp. 40-49, 2006.

[10] T.K. Buennemeyer, "Battery-Sensing Intrusion Protection System (B-SIPS)," Doctoral Dissertation, Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg,VA, 2008.

[11] LeCroy, "Merlin II Analyzers," http://www.lecroy.com/tm/products/ProtocolAnalyzers/MerlinII.asp?menuid=60, 2008.

[12] Remote-Exploit, "Backtrack 3," http://www.remote-exploit.org/backtrack_download.html, 2008.