10 | 18 Sonderdruck für CryptoMagic

IT Administrator
Das Magazin für professionelle System- und Netzwerkadministration

Technik und Risiko von Spectre und Meltdown

www.it-administrator.de

Technik und Risiko von Spectre und Meltdown

Warten auf den Weltuntergang

von Markus Schräder

Pünktlich zu Weihnachten 2017 erschienen Artikel mit Aufmachern vom SuperGAU oder Atomexplosionen im Prozessor. Der IT-Security-Weltuntergang durch die Spectre- und Meltdown-Sicherheitslücken schien unabwendbar, da diese Bugs Angriffsflächen für Hacker in so gut wie jeder aktuellen Hardware bieten. Ein halbes Jahr später fragen wir uns, worum es da eigentlich ging und warum manche immer noch auf den Weltuntergang warten, während andere die Ereignisse entspannt ignorieren.

In der Anfangszeit der Intel-x86-CPUs hatte jede Software vollständigen Zugriff auf den gesamten Speicher der Hardware. Software musste in diesem sogenannten "Real-Mode" absolut fehlerfrei arbeiten, damit ein stabiles Gesamtsystem zustande kam, denn jeder Fehler hatte oft genug einen Systemabsturz zur Folge. Um dies grundlegend zu ändern, wurde eine Trennung der Speicherbereiche pro Software über das sogenannte "Virtual Memory" im "Protected Mode" eingeführt.

Virtual Memory leitet hardwarebasiert jeden Speicherzugriff einer Software auf einen eigenen und zunächst leeren virtuellen Speicher um. In diesem lässt sich an beliebiger Stelle echter Speicher hinterlegen, um der jeweiligen Software den Zugriff hierauf zu ermöglichen. Jeder Zugriff auf leeren, also nicht hinterlegten Speicher schlägt fehl und führt standardmäßig zum kontrollierten Abbruch der Ausführung der jeweiligen Software. Auf diesem Wege, gesteuert über sogenannte Ringe [1], ist es einem Betriebssystem möglich, seinen eigenen Speicher sowie den Speicher von Software vor dem fehlerhaften Zugriff einer anderer Software zu schützen und bei Fehlern einer Software die ungewollten Auswirkungen auf diese einzudämmen oder zu verhindern. Der Protected Mode wurde mit dem Ring "-1" so erweitert, dass dieser Mechanismus nicht nur zwischen Software funktioniert, sondern auch zwischen virtualisierten Betriebssystemen. Mit diesen Funktionen stellt der Protected Mode den zentralen Grundpfeiler der Stabilität der x86-Plattform dar.

Sicherheitsversagen des Protected Mode

Die aktuellen Meltdown- und Spectre-Angriffe funktionieren nach einem ähnlichen Prinzip wie die "Beweisführung" der beliebten TV-Figur Inspektor Columbo: Durch geschicktes Nachfragen versucht der Inspektor von seinem Verdächtigen ein Verhalten zu provozieren, das ihn letztendlich als Schuldigen entlarvt, ohne dass sich dieser dessen bewusst ist. Die Verdächtigen reagieren hierbei auf scheinbar unverfängliche Fragen, sodass sich kombiniert ein überführendes Bild ergibt.

Moderne Prozessoren sind stark auf Performanz optimiert, beispielsweise durch schnelle Zwischenspeicher (Caches) oder durch das spekulative, vorzeitige Ausführen von Software. Diese Optimierungen führen zu Nebenwirkungen, die in Kombination nur schwer zu vermeiden oder sicher einzudämmen sind. So wie die Verdächtigen in der Fernsehserie Columbo den Tathergang unbeabsichtigt verraten, so führen Nebenwirkungen in Prozessoren zu der Gefahr, dass diese genutzt werden können, um Daten zu lesen. Meltdown stellt einen solchen Fall dar: Durch das Messen von Speicherzugriffszeiten können

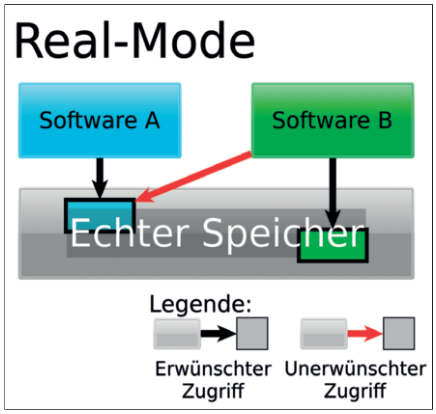

Bild 1: Früher konnte jede Software immer auf alles zugreifen, sodass hier Meltdown oder Spectre gar keinen Angriff dargestellt hätten.

in Kombination mit Nebenwirkungen, die sich durch die spekulative Ausführung von Software ergeben, Daten gelesen werden. Da alle bisherigen neuartigen Angriffe im Spectre- und Meltdown-Umfeld auf mindestens einer Nebenwirkung basieren, die sich erst durch die spekulative Ausführung von Software ergibt, sehen viele Experten diese Optimierung als zentrale Ursache für die Gesamtproblematik an.

Diese neuartigen Angriffe überwinden mittels der Nutzung von Nebenwirkungen den Stabilitätsgrundpfeiler der x86-Plattform: die Speichertrennung des Protected Mode. Von diesen Angriffen sind auch die ARM- und PowerPC-Plattform betroffen, somit sind neben Workstations und Serversystemen auch NAS-Geräte und Smartphones (iOS und Android) gefährdet. Der hardwareunterstützte Speicherschutz des Protected Mode lässt sich durch Software überwinden. Hierbei werden weder Spuren hinterlassen noch ist dieser Angriff auf Fehler in Betriebssystem oder anderer Software angewiesen. Eine Angriffssoftware muss lediglich regulär auf der Hardware ablaufen, um durch kombinierte, selbst herbeigeführte Nebenwirkung diese Angriffe auszuführen. Hierbei stellt auch die Grenze von einem virtuellen Betriebssystem zu einem anderen teilweise keine Hürde dar.

Die Umgehbarkeit des Speicherschutzes des Protected Mode über Meltdown oder Spectre mag in bestimmten Umgebungen überschaubare Risiken haben, wenn wir bedenken, dass hierfür Schadsoftware auf dem System ausgeführt werden muss. Sofern es sich um Einzelbenutzersysteme handelt, kann Malware auch ohne Meltdown und Spectre erhebliches Schadenspotenzial entfalten. Spätestens im Rahmen von mehreren gleichzeitig virtualisierten Betriebssystemen und/oder bei unbekanntem Umfang des Einsatzes von vertrauensunwürdiger Software führt dies möglicherweise zur Verletzung der Vertraulichkeit beziehungsweise zur Kompromittierung von Systemen und Daten.

Neue Angriffsvarianten

Neuere Varianten der Angriffe sollen neben lesenden auch schreibende Zugriffe ausführen können und zudem auch CPU-interne Daten (Register) erreichen [2]. Letzteres würde beispielsweise das Auslesen geheimer Schlüssel der CPU-Beschleunigungserweiterung zur AES-Verschlüsselung (AES-NI) ermöglichen [3].

Seit Kurzem sind acht neue Angriffe bekannt (Spectre-NG), die neue und bislang unbekannte Wege für funktionierende Attacken zeigen sollen. Deren Details werden jedoch derzeit zum Teil noch geheim gehalten [4]. Mit "Foreshadow" soll ein neuer Meltdown-ähnlicher Angriff speziell gegen Virtualisierungsinfrastrukturen [5] möglich sein. Zudem sollen unter Umständen Attacken auch über das Netzwerk funktioniert [6]. Das volle Ausmaß ist aufgrund der unvollständigen Informationen derzeit nicht absehbar.

Vergessene Gefahr

Die Grundlage der Spectre- und Meltdown-Angriffe, über Software interne Strukturen des Prozessors anzugreifen, wurde bereits vor Jahrzehnten beschrieben. Speziell zur Sicherheit der Intel-80x86-Prozessorarchitektur wurden von der amerikanischen "Oxford Systems Inc." sowie "The Aerospace Corporation" in Zusammenarbeit mit der amerikanischen National Security Agency in einem wissenschaftlichen Artikel acht Schwachstellen beschrieben, die Einfluss auf sicherheitskritische Daten innerhalb des 80386- und 80486-Prozessors haben.

Obwohl diese Anfälligkeit der Prozessoren über Jahrzehnte bekannt war, hat sich hierfür offensichtlich nie ein Bewusstsein entwickelt. Da neben der x86-Plattform auch mindestens eine andere Plattform ähnlich betroffen ist, scheint dies nicht nur für Erstere zu gelten. Es lässt sich die Vermutung nicht ganz von der Hand weisen, dass von der Stabilität des Protected Mode ohne weitere Prüfung auf einen verlässlichen Angriffsschutz desselben geschlossen wurde, da es kaum einen Anlass gab, die Robustheit gegen gezielte Angriffe infrage zu stellen.

Andererseits ist zu berücksichtigen, dass der direkte Zugriff auf Daten, lesend wie schreibend, über Prozessgrenzen hinweg in etablierten Betriebssystemen auch regulär zur Laufzeit vorgesehen ist [7, 8, 9]. Auch über das Dateisystem kann entweder direkt auf Daten zugegriffen werden oder eine Software wird vor der Ausführung kopiert und verändert ausgeführt. Dies ist insbesondere für macOS ein wichtiger Aspekt, da hier für den Speicherschutz seit einigen Jahren engere Anforderungen gelten. Das bedeutet, dass selbst wenn es Spectre und Meltdown nicht gäbe, trotzdem der gezielte Angriff von einer Software auf eine andere Software möglich ist. Erschwerend kommt hinzu, dass die aktuellen Angriffe vom internen Aufbau des jeweiligen Prozessors abhängig sind, also von zumindest nicht öffentlich verfügbarer Dokumentation oder vom aufwendigen und langwierigen

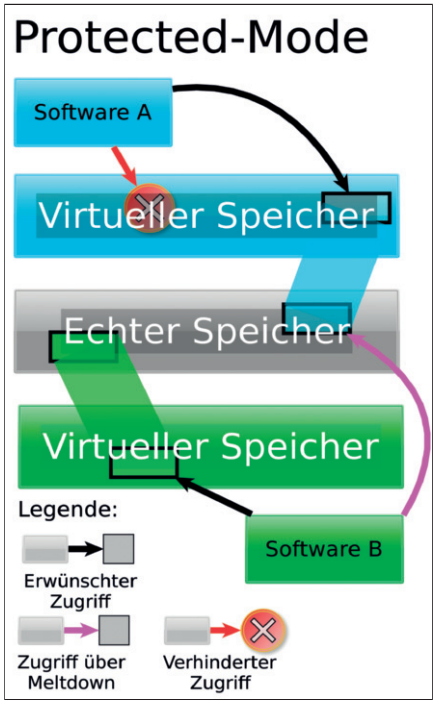

Bild 2: Virtual Memory im Protected Mode und die Wirkung eines Meltdown-Angriffs.

Reverse-Engineering. Als weitere Unsicherheitsfaktoren für einen erfolgreichen Angriff sind mögliche Serien- und Modellunterschiede der Prozessoren sowie Abhängigkeiten zu anderen Hardwarekomponenten, wie beispielsweise dem Speicher und dessen Timings, zu nennen. Diese Voraussetzungen und Aufwandsbereitschaft sind kaum woanders als in der Forschung oder beim Hersteller des jeweiligen Prozessors zu erwarten.

Wenig bekannte Hintertüren

Um dieses Ausnutzen von Nebenwirkungen im Prozessor abzuwenden, ohne sich auf die im Fluss befindlichen Microcode- oder Betriebssystemupdates verlassen zu müssen, bleibt dem IT-Verantwortlichen nichts anderes übrig, als die Ausführung von Software aus nicht vertrauenswürdiger Quelle grundsätzlich zu verhindern. Dies mag auf den ersten Blick einfach erscheinen, doch spätestens seit dem Kursieren von Crypto-Trojanern sollte keine leichtfertige Ausführung von nicht vertrauenswürdiger Software mehr stattfinden.

So wähnt sich der IT-Verantwortliche am Ziel einer einfach zu erreichenden Spectre- und Meltdown-sicheren Umgebung, es gibt allerdings Stellen der Ausführung von nicht vertrauenswürdiger Software, die nicht einfach zu identifizieren sind. Ein einfaches Beispiel hierfür ist der Test von Software in Virenscannern zur Verhaltensanalyse. Hierzu muss der Virenscanner die nicht vertrauenswürdige Software auf der Hardware ausführen, ohne dass dies für den Benutzer offensichtlich ist. Noch versteckter ist der bislang unproblematische Ansatz, wenn nicht vertrauenswürdige Software in einer sicheren Umgebung (Sandbox) beziehungsweise mit reduziertem Befehlsumfang ausgeführt wird. Die in den meisten Browsern integrierte Skriptsprache JavaScript ist ein relativ einfach zu identifizierendes Beispiel, das die Browserhersteller mittlerweile im Griff zu haben scheinen [10]. Auch PDF-Betrachter und das Flash-Plug-in sind bekannt dafür, Skriptsprachen und/oder Sandboxing zu unterstützen. Diese Praxis ist bei jeder Software, die externe Daten verarbeitet, ein potentielles Sicherheitsproblem. Nicht nur Webseitenbetreiber können in potenzielle Sicherheitsprobleme laufen, wenn Softwarekomponenten genutzt werden, die Skriptsprachen oder Sandboxing unterstützen.

Die Frage der Angreifbarkeit stellt sich in Multiuser- und Virtualisierungsumgebungen noch grundlegender. Während der einzelne Nutzer einer Hardware verhältnismäßig gut eingrenzen kann, welche Software ausgeführt wird, so wird dies hier oft unmöglich. Üblicherweise schließen derartige Systeme die Möglichkeit mit ein, dass Dritte potenziell nicht vertrauenswürdige Software ausführen: Besonders "in der Cloud" und bei der Anmietung von virtuellen Servern liegt keine Information darüber vor, welche Software in anderen virtuellen Betriebssystemen auf derselben Hardware läuft. Da dies als möglicher Angriffsvektor von Meltdown oder Spectre zu sehen ist, sollten IT-Verantwortliche das Risiko genau abwägen. Denn weder der Nutzer noch der Anbieter hatten bisher im Normalfall im Blick, von wem welche andere Software gleichzeitig auf einer geteilten Hardware ausgeführt wird. Folglich herrscht auch Unkenntnis darüber, wer diese neuen Angriffe auf dort laufende Software durchführen kann. Für deutlich mehr Sicherheit sorgt hier dedizierte Hardware.

Auch Smartphones betroffen

Da die ARM-Plattform ähnlich betroffen ist, stellen sich für die meisten Smartphones grundsätzlich die gleichen Fragen wie bei der x86-Plattform. Laut ARM sollen jedoch keine neuen CPUs mehr von diesen Angriffen über die spekulative Ausführung von Software betroffen sein [11]. Auch wenn üblicherweise nur eine Person ein Smartphone nutzt, so ist die Software selten vom Benutzer selbst hergestellt. Diese wird üblicherweise über App-Stores heruntergeladen und parallel ausgeführt. Hierbei sind die gleichen Probleme wie bei Mehrbenutzerumgebungen auf der x86-Plattform gegeben. Da über Updates der jeweiligen App jederzeit neue Software auf das System kommt, gelangen über diesen Weg auch potenziell neue Angriffe auf das Gerät.

Leider ist beim weit verbreiteten Android der jeweilige Hersteller für Updates zuständig, die dieser meist nur unregelmäßig ausliefert – wenn überhaupt. Fraglich ist zudem, ob diese eher hardwarelastigen Hersteller derartige Probleme überhaupt erschöpfend adressieren und Updates des Betriebssystems sowie mögliche Microcode-Updates des Prozessors ausliefern können. Für heutige Android-Geräte ist folglich davon auszugehen, dass viele von diesen dauerhaft anfällig bleiben. Etwas besser sieht es hier bei iOS aus, bei dem es nur einen Hardwarehersteller gibt. Zumindest jüngere Geräte wurden innerhalb weniger Wochen nach dem Bekanntwerden der Angriffe mit einem Update versorgt, das Spectre-Probleme softwareseitig beseitigen soll.

Das Risiko minimieren

Es gibt deutliche Hinweise darauf, dass insbesondere Hyperthreading einige Angriffsvarianten begünstigt. OpenBSD-Entwickler diskutieren aus diesem Grund, Hyperthreading standardmäßig zu deaktivieren [12]. Die Angreifbarkeit von Skriptsprachen wurde reduziert, indem dem laufenden Skript keine präzisen Zeitangaben mehr zur Verfügung gestellt werden, denn Timing-Analysen werden infolge der Unschärfe im Millisekunden-Bereich unbrauchbar [10]. Auch lassen sich Angriffe zwischen virtualisierten Betriebssystemen teilweise verhindern, wenn diese im 32-Bit-Modus laufen [13].

Gegen den Meltdown-Angriff bietet das Betriebssystem die Möglichkeit, den Kernel durch "kernel page-table isolation" (vormals KAISER) vor Angriffen zu schützen. Der Kernelspeicher in Linux, Windows und macOS ist in den jeweils aktuellen Versionen hierdurch weitestgehend geschützt. Bestimmte Spectre-Angriffe lassen sich zudem eindämmen, wenn die angegriffene Software spezielle CPU-Befehle vor relevanten Aktionen aufruft. Diesen ohnehin Microcode-abhängigen Schutz integriert der Compiler während der Übersetzung automatisch: Dies führt jedoch unter Umständen zu solch massiven Performanzverlusten, dass die wenigsten Softwarehersteller den Schutz einsetzen [14].

Inwieweit die Microcode-Updates der Prozessoren umfänglich helfen, lässt sich leider aufgrund der unvollständigen Informationen zu den Angriffen noch nicht überprü-

fen. Einige Updates wurden zurückgezogen, da diese zu Problemen führen, zudem ist für ältere Prozessoren nicht vorgesehen, entsprechende Updates auszuliefern. In den Betriebssystemen haben die Hersteller Entschärfungen integriert, sich eine Übersicht hierzu zu verschaffen, ist nicht trivial und die Integration von Maßnahmen ist noch nicht abgeschlossen.

Es ist möglich, Software zu schreiben, die anfällige Systeme erkennt und somit Handlungsbedarf aufzeigt. Eine wirklich sichere Möglichkeit, immune Systeme zu erkennen, existiert aber nicht: Wenn eine entsprechende Software ein System als sicher deklariert, bedeutet das leider keine Entwarnung. Erst mit dem Einsatz der für die zweite Jahreshälfte 2018 angekündigten CPUs ist eine echte Immunität zu erwarten, da erst diese auf einem neuen Hardwaredesign beruhen und nur auf diesem Wege hardwareseitige Fehler grundsätzlich beseitigt werden können.

Das praktische Risiko bewerten

Die Schadenshöhe eines unerlaubten Zugriffs auf Daten ist vom Schadenspotenzial des Bekanntwerdens selbiger abhängig – eine einzelne Arbeitsstation, Notebook oder Smartphone hat hierzu meist weniger zu bieten als ein Cloudsystem mit möglicherweise hunderten von unterschiedlichen Nutzern. Natürlich kann auch das Auslesen von Passwörtern indirekt zum Verändern von Daten führen, im Vergleich zu anderen möglichen Angriffen stellt die Nutzung von Meltdown oder Spectre jedoch in den meisten Fällen einen vergleichsweise sehr hohen Aufwand dar.

Die Eintrittswahrscheinlichkeit ist wesentlich schwieriger zu beurteilen. Auch wenn die verfügbaren wissenschaftlichen Arbeiten danach klingen, als ob unmittelbare Gefahr droht, können wir nicht bestätigen, dass die Angriffe ohne weiteres umsetzbar sind und eine unmittelbare Bedrohung darstellen. Zumindest für den Meltdown-Angriff haben wir uns dies genauer angeschaut: Der von Michael Schwarz vorliegende Proof of Concept [15] basiert darauf, dass ein speziell konstruiertes anzugreifendes Programm seine Daten in einer Endlosschleife liest, diese somit im CPU-Cache verbleiben und sich diese Daten von einer Angriffssoftware auslesen lassen. Ohne die Schleife in der angegriffenen Software funktioniert der Angriff auf keiner von uns getesteten Hardware (moderne Intel-Prozessoren). Zudem muss das angreifende Programm die Speicheradresse, an der sich die zu lesenden Daten befinden, erfahren. Im Proof of Concept wird diese Information vom "Opfer" verraten, ein realer Angriff müsste diese Information zusätzlich gewinnen oder den gesamten Arbeitsspeicher nach der Zielinformation durchsuchen. Googles Project-Zero-Bericht geht davon aus, dass eine L1-Cacheabhängigkeit für Meltdown besteht [16], was unsere Beobachtungen bestätigen.

Auf Nachfrage bei den Autoren des Meltdown-Papers [17], unter welchen Umständen (Hard- und Softwarekonfiguration) ein Auslesen des Speichers von herkömmlicher Software, die kein Vorhalten der Daten im Cache provoziert, mit 503 KByte/s [18] möglich ist, erhielten wir lediglich den Verweis auf verschiedene Videos, die als Nachweis dienen sollen. Leider konnten wir die postulierten Datenraten nicht ansatzweise nachvollziehen: Mehr als wenige Byte/s ließen sich nicht lesen – und auch nur an bekannter Speicherstelle. Die Nachweiskraft sowie die Aufklärung von offensichtlichen Widersprüchen in Videos, die unter unveröffentlichten Rahmenbedingungen zustande gekommen sind, sollen an dieser Stelle nicht diskutiert werden.

Fazit

Die ersten Veröffentlichungen – und somit auch die ersten Code-Beispiele – zu Meltdown und Spectre liegen mehr als ein halbes Jahr zurück, ohne dass erfolgreiche Angriffe bekannt wurden. Bedenken wir, in welch kurzen Zeiträumen Black-Hat-Hacker üblicherweise weit weniger fatale Lücken ausnutzen, ist dies überaus verwunderlich. Dies und die enormen Schadpotenziale, die Meltdown und Spectre angeblich haben, lassen nur zwei Schlüsse zu: Entweder das Risiko ist weit geringer oder die Attacken erfordern derart große Ressourcen, dass sie nicht lohnenswert erscheinen. Infolge der unvollständigen Informationslage lässt sich derzeit hierzu keine genaue Aussage treffen, genauso lässt sich nur erahnen, welche Aktivitäten bei Geheimdiensten hierzu laufen. *(jp)*

Markus Schräder ist Geschäftsführer der CryptoMagic GmbH.

Link-Codes

[1] Ring (CPU)
i0z71

[2] Speculative Buffer Overflows: Attacks and Defenses
i0z72

[3] Exploiting lazy FPU state switching
i0z73

[4] Super-GAU für Intel
i0z74

[5] Details and Mitigation Information for L1 Terminal Fault
i0z75

[6] NetSpectre: Read Arbitrary Memory over Network
i0z76

[7] Direkter Zugriff auf Daten über Prozessgrenzen hinweg (macOS)
i0z77

[8] Direkter Zugriff auf Daten über Prozessgrenzen hinweg (Linux)
i0z78

[9] Direkter Zugriff auf Daten über Prozessgrenzen hinweg (Windows)
i0z79

[10] What Spectre and Meltdown Mean For WebKit
i0z70

[11] Vulnerability of Speculative Processors to Cache Timing Side-Channel Mechanism
i0z7a

[12] Simultanious Multi Threading
i0z7b

[13] Xen Project Spectre/Meltdown FAQ
i0z7c

[14] Spectre Mitigations in Microsoft's C/C++ Compiler
i0z7d

[15] Several applications demonstrating the Meltdown bug
i0z7e

[16] Reading privileged memory with a side-channel
i0z7f

[17] If a process is not accessing the memory to be attacked in an endless loop all the time, the memory is just not readable by meltdown
i0z7g

[18] Meltdown
i0z7h