



μ-Xen

Ian Pratt
SVP, Products
Bromium Inc.

- μ -Xen Goals
- Architecture and implementation
- Evolutionary path
- Use case: Bromium end-point security

The μ -Xen Client Hypervisor



- Derived from the Xen code base
- Designed to address client hypervisor deployment challenges
- Designed to achieve a high-degree of security assurance
- No requirement to support legacy s/w or h/w
- Optimized to support *micro-virtualization*
- Evolution to a new class of hypervisor

- Start with a self-contained type-2
 - Host kernel module containing μ -Xen blob
 - Well-defined interface to/from the host
 - Memory allocation, Schedule VCPUs, Timers, Events
- Similar to hXen project
 - Windows, OS X; easily portable
 - μ -Xen drives VT hardware, leaves IO and other hardware to host OS

μ-Xen streamlines Xen code base



- x64 only
- No support for systems that are not VTx/EPT or AMDV/NPT capable
- No support for PV MMU guests
- Significant code size reductions
- No compat code
- Separate build target
- Future: slim down instruction emulator

- μ-Xen demultiplexes IORecs and events to multiple host processes per VM
 - Able to have separate video, audio, disk, net etc processes
 - Helps scheduling latency, enables better exploitation mitigation lock-down
- Use just the relevant parts of QEMU
- User-space PV backend drivers
- Post-boot virtual IO device revocation

Support for large numbers of similar VMs



- Support for VM fork
 - Simple extension of Populate on Demand code
- Tree of templates/VMs supported
- PV driver support for re-sharing free pages
- IO tracking to re-share swapped and re-read pages

- Goal: host is responsible for resource allocation, but can not interfere with the privacy or integrity of the hypervisor or other guests, e.g.:
 - Hypervisor asks host for memory, receives it, then removes host access to it
 - Host may perform IO on behalf of hypervisor and guests, but must be privacy and integrity protected
 - Host may deny service, crash; IO tamper detectable

De-privilege host into a VT container



- Remove host access to hypervisor and guest memory in EPT tables
 - h/w devices passed through to host, VTd protected
- Device models can only access Granted memory
 - Require PV network and disk as no virtual DMA
 - Guest IO buffers encrypted/authenticated before grant copy/map
- Save/restore, dom builder must first have guest memory encrypted/authenticated by hypervisor
 - Hypervisor paging requires similar treatment

- Migrate certain host h/w devices to ownership by hypervisor or privileged guest(s), replace with virtual device
 - E.g. Keyboard controller to avoid key logging
 - Possibly WiFi/NIC too
- Measured launch of hypervisor
 - TXT/TPM to establish DRoT
 - Remote attestation of hypervisor configuration



Task-based Micro-virtualization



The Challenge

Br



The Register®

Hardware Software Music & Media Networks Security Cloud Public Sector Business Science

Crime Malware Enterprise Security Spam ID

Print Comment Tweet Like 17 Alert

NASA lost 'full control' to hackers, pwned 13 times last year Houston still has a problem with security

By [John Leyden](#) • [Get more from this author](#)

Posted in [Enterprise Security](#), 5th March 2012 15:17 GMT

Cybercrooks broke into NASA's computer systems 13 times last year gaining "full functional control" of important systems in the worse cases, according to the testimony before the US Congress by the space agency's inspector general.

Paul Martin told a Congressional panel on information security at the space agency that NASA spent \$58m of its \$1.5bn annual IT budget on cyber security. The space agency has long been a prestige target for hackers of various skill levels and motivations, including profit-motivated malware distributors (cybercrooks) and intruders thought to be in the pay of foreign intelligence services.

RSA hack ta

By [Jack Clark](#) (@mappingbab

About this blog

The computer security company has disclosed.



STAFF

Mapping Babel

On Friday, Uri Rivner, RSA's head of new technologies for consumer identity protection, detailed the methods used to penetrate RSA. The attack, which RSA disclosed on March, saw hackers steal information about RSA's SecureID authentication tokens, which are used to perform two-factor authentication for users of various networks.



What Went Wrong?



- Poor Isolation leads to Vulnerability
 - Multi-tenancy of code (& data) from a huge set of trust sources
 - Code & Data are executed in the context of a few, weakly organized and poorly isolated “bins” representing levels of trust - e.g. UID, GID, PID
 - Overloaded, complex specs & buggy implementations of porous interfaces
- Thus malware can:
 - Access data and documents in the same “bin”
 - Subvert other good code that is placed into the same “bin”

Why Legacy Security Solutions Fail

Br

Black-listing: **Detection is not protection**

- Doesn't protect users from zero day attacks
- Unable to detect targeted & advanced malware

White-listing: **Restriction is not protection**

- Users want to be productive
- Attackers can compromise trusted applications and web sites

The Challenge

A person in a wetsuit is sitting on a blue surfboard in the ocean, looking towards a large, dark green wave. The sky is a pale, hazy blue. The overall scene conveys a sense of challenge and uncertainty.

How do we plan for the unknown threat?

What do we do about human mistakes?

The Micro-Virtualization Approach

Br

Next generation endpoint security

- A pragmatic approach to trusted computing

Based on hardware based isolation and not detection

- Contain risky activity from corporate data and network
- Block zero day and polymorphic threats

Provides isolation & trust through hardware virtualization

Micro-virtualization



Virtualizes vulnerable tasks within a single Windows desktop

Lightweight, fast, hidden, with an unchanged native UX

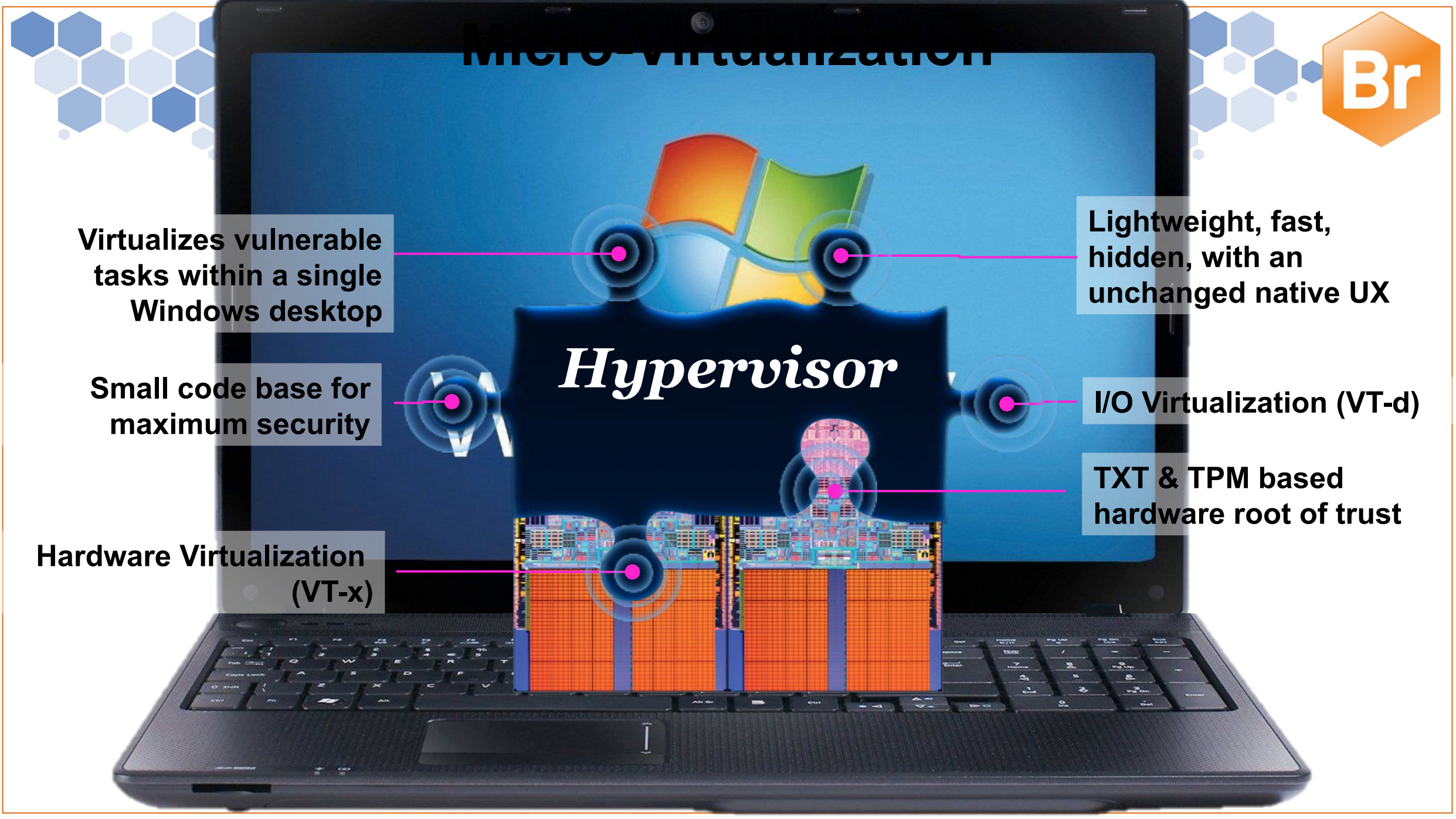
Small code base for maximum security

I/O Virtualization (VT-d)

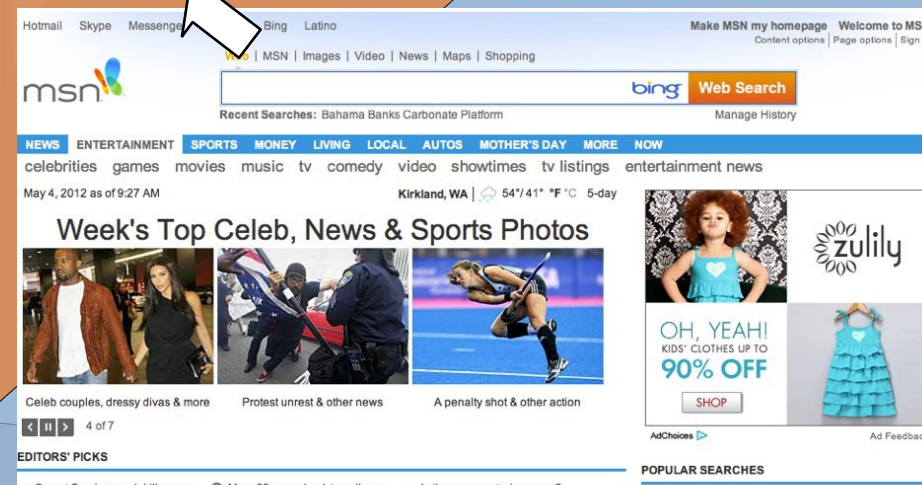
Hypervisor

TXT & TPM based hardware root of trust

Hardware Virtualization (VT-x)



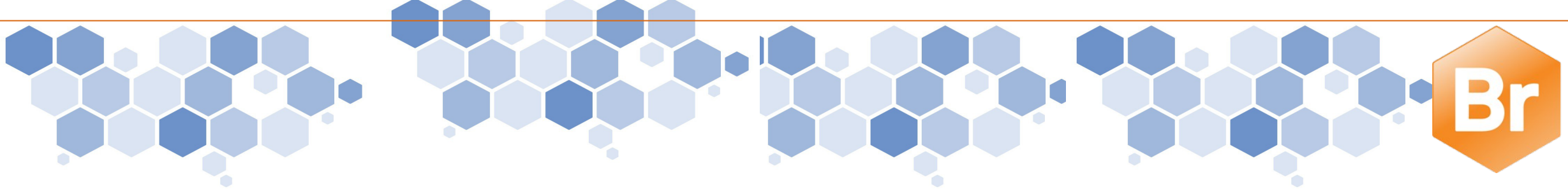
The Microvisor isolates vulnerable tasks from Windows, each other & key system resources



Micro-VMs have "least privilege" access to files, networks & devices, and execute CoW

Each vulnerable task is instantly isolated in a micro-VM, invisible to the user





- Demo

Key Threat Vectors Are Isolated

Web Browsing
External Documents



E-mail Attachments
USB files



- μ -Xen is targeted at client systems
- Combines ease of deployment with useful security properties
- Enables task-based micro-virtualization, allowing mandatory access control to be retrofitted to commodity client OSes
- **We're hiring! Please email ian@bromium.com**