PrivExec

Private Execution as an Operating System Service

Kaan Onarlioglu, Collin Mulliner,
William Robertson and Engin Kirda

2013 IEEE Symposium on Security and Privacy

Intro

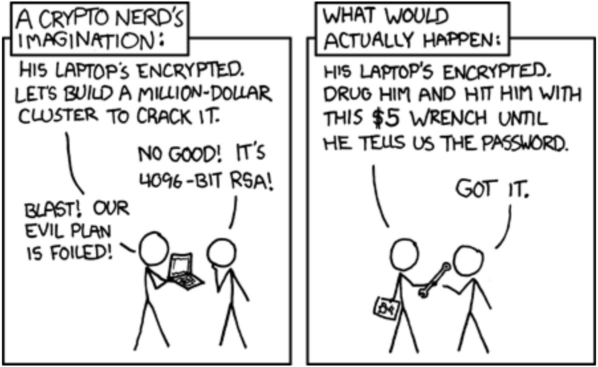
Observations

- Privacy gains importance
- Wiping data on disk/“Private Browsing” → Unreliable
- Full-disk encryption → Coercion

Intro

Observations

- Privacy gains importance
- Wiping data on disk/“Private Browsing” → Unreliable
- Full-disk encryption → **Coercion**


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

Source: `http://xkcd.com/538/`

Intro

Observations

- Privacy gains importance
- Wiping data on disk/“Private Browsing” → Unreliable
- Full-disk encryption → Coercion

Threat Model

- Benign applications
- Phase 1 – Execution: Normal user with remote access
- Phase 2 – Session ended: Physical access

Design

Goals

- Data from a private execution is never leaked
- Secure disposal of private data after termination
- No cooperation required from application or filesystem
- Flexibility

Design

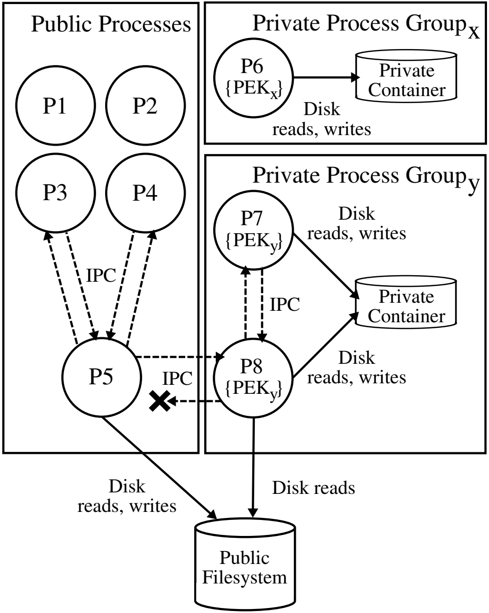
Goals

- Data from a private execution is never leaked
- Secure disposal of private data after termination
- No cooperation required from application or filesystem
- Flexibility

Private Process Group

- Bound to ephemeral *private execution key* (PEK)
- *Secure storage container*
- Partitioned swap space
- Restricted IPC

Design Overview



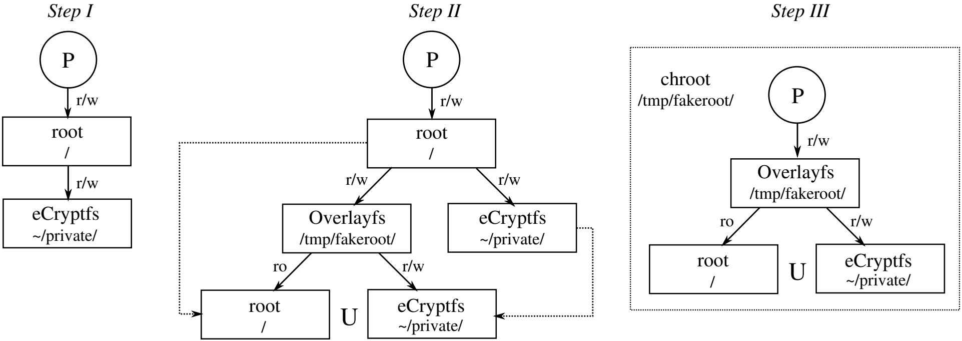
Implementation

- PEK stored in process descriptor (kernel memory) and inherited by children
- modify process management (`do_fork`, `do_exit`)
- modify paging (`pageout`, `do_swap_page`) using *Crypto API*
- secure storage container: *eCryptfs* + *Overlayfs*
- Wrapper to run ordinary application in private mode

Implementation

- PEK stored in process descriptor (kernel memory) and inherited by children
- modify process management (`do_fork`, `do_exit`)
- modify paging (`pageout`, `do_swap_page`) using *Crypto API*
- secure storage container: *eCryptfs* + *Overlayfs*
- Wrapper to run ordinary application in private mode
  1. Create private copy of itself
  2. Setup secure storage container
  3. Load application in `chroot`
  4. Clean up

Setting Up The Secure Storage Container



Disk I/O And Filesystem Performance

	Original	eCryptfs-only		PrivExec	
	Performance	Performance	Overhead	Performance	Overhead
Write	110694.60 KB/s	97536.83 KB/s	13.49 %	97979.47 KB/s	12.98 %
Rewrite	48724.53 KB/s	38800.78 KB/s	25.58 %	38790.07 KB/s	25.61 %
Read	111217.67 KB/s	107134.53 KB/s	3.81 %	106293.73 KB/s	4.63 %
Seek	196.27 seeks/s	147.53 seeks/s	33.04 %	138.37 seeks/s	41.84 %
Create	13906.73 files/s	8312.73 files/s	67.29 %	8181.10 files/s	69.99 %
Stat	217734.60 files/s	126326.23 files/s	72.36 %	117844.75 files/s	84.76 %
Delete	42012.87 files/s	25232.67 files/s	66.50 %	23017.00 files/s	82.53 %

Runtime Performance Overhead I

	Firefox		
	Orig. Runtime (s)	PrivExec Runtime (s)	Overhead
Alexa	98.43	103.56	5.21 %
Wikipedia	37.80	39.96	5.71 %
CNN	66.61	69.15	3.81 %
Gmail	58.43	61.36	5.02 %

Chromium		
Orig. Runtime (s)	PrivExec Runtime (s)	Overhead
91.63	94.69	3.34 %
39.25	40.12	2.22 %
49.21	50.83	3.29 %
30.61	30.98	1.21 %

Runtime Performance Overhead II

	Orig. Runtime (s)	PrivExec Runtime (s)	Overhead
Audacious	61.27	62.30	1.68 %
Feh	51.86	52.52	1.27 %
FFmpeg	105.47	111.31	5.54 %
grep	245.37	253.82	3.44 %
ImageMagick	96.16	101.41	5.46 %
LibreOffice	99.64	100.62	0.98 %
MPlayer	122.98	129.39	5.21 %
Pidgin	116.49	117.87	1.19 %
Thunderbird	75.45	78.78	4.41 %
Wget	71.48	71.89	0.57 %

Conclusion

Summary

- Few modifications of Linux
- Runs existing applications
- Small ($< 6\%$, 3.31% avg) impact on performance
- Safe according to threat model

Limitations

- System hibernation
- Priviledged users
- X applications

⇒ Code available at `http://www.onarlioglu.com/privexec/`

Discussion

- How does encryption of swapped pages work?
- Does privacy really gain importance?
- Usability? (e.g. downloads)
- Bugs?