HACKTALKS

PRESENTS

Kernel Cat™

By

Gil Dabah & Tomer Teller

HACKTALKS PRESENTS

Kernel Cat™

By

Gil Dabah & Tomer Teller

ZAUBER BROTHERS™ 1984

NORTHBIT

ABOUT – GIL DABAH

- CEO of NorthBit
- TinyPE Challenge
- Patchehd IE VML bug [ZERT] (Faster than MSFT, don't tell Tomer)
- diStorm Disassembler Library
- Published 0d's in Windows Kernel → kernel lover

ABOUT – TOMER TELLER

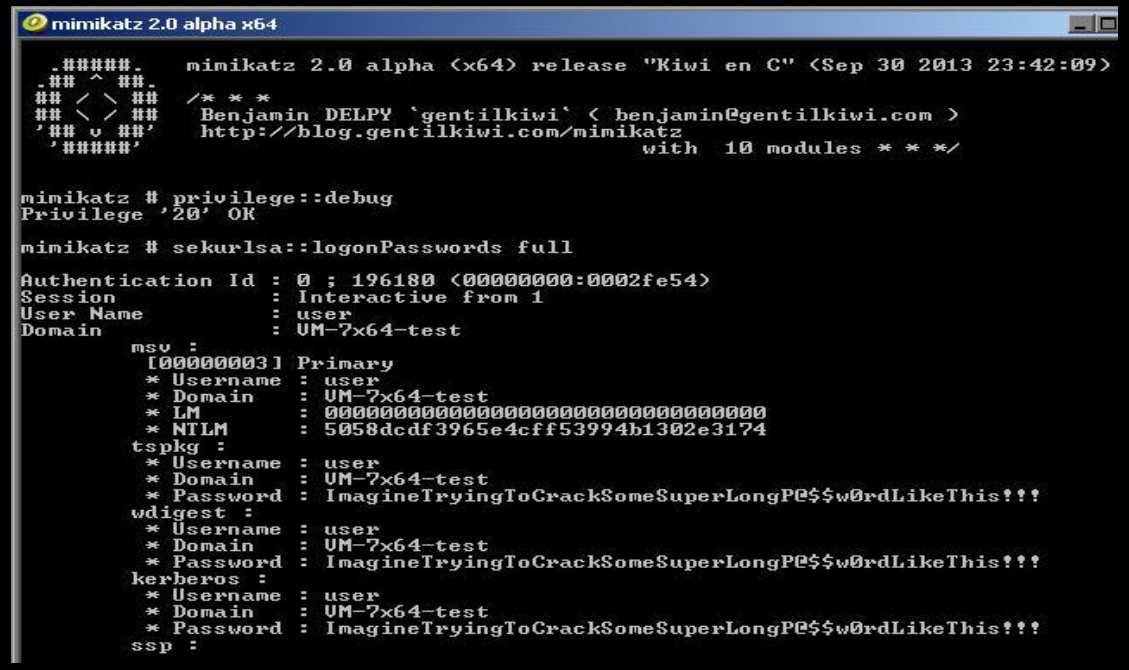
- Microsoft Azure Cybersecurity
- 10 years at Check Point (security innovation research manager)
- 5+ patents exploit mitigation field
- Speaker at Blackhat, RSA and OWASP
- Cat lover

ABOUT – KERNEL CAT

- Born in 1984
- Alley Cat’s evil brother
- Loves to hack computer networks
- Lateral Movement with Mimikatz

MIMIKATZ

- Windows Post exploitation tool
- Extract sensitive information from memory
  - Plaintext passwords (pre windows 8)
  - Password hashes
  - Kerberos tickets
- Pass-The-Hash/Ticket



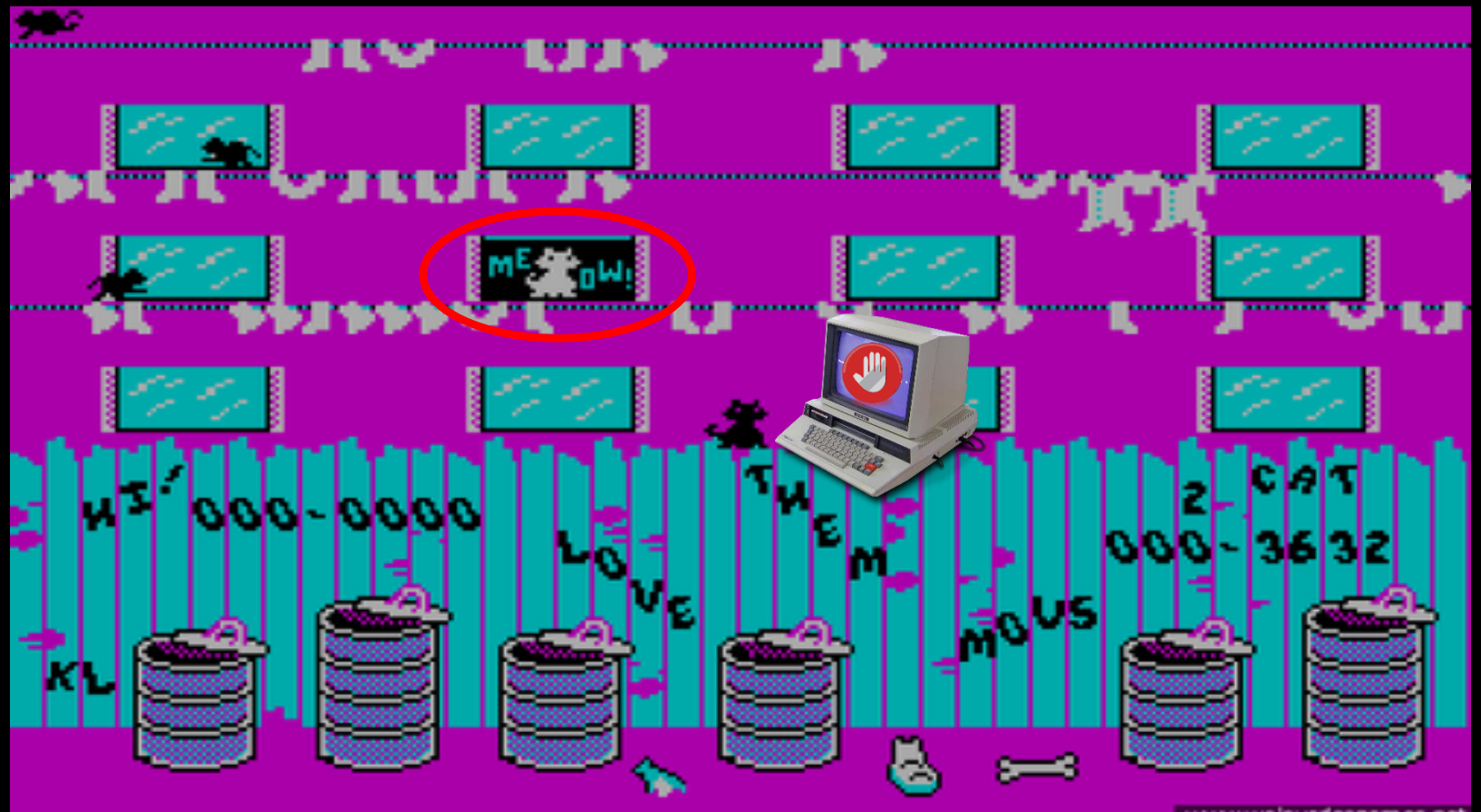
LSASS

LOCAL SECURITY AUTHORITY SUBSYSTEM SERVICE

- Responsible for enforcing the security policy on the system
- Verifies user login onto a Windows computer
- Manages the SAM file DB for local users
- LSASS stores all the hashes in memory of logged in users
  - Supports the Single-Sign-On process

ONCE UPON A TIME...

- Kernel cat hacked a computer network but **failed** to reach his cat lady
  - Mimikatz failed to dump the admin password hash from LSASS



ONCE UPON A TIME...

- Kernel cat hacked a computer network but **failed** to reach his cat lady
- The computer he hacked into was running a hardened Windows 8.1
  - LSASS runs as a **Protected Process Light** (“the admin everywhere problem”)
  - Any user privilege including SYSTEM cannot access this process (in usermode)
  - Plain text passwords are no longer obtainable
  - Password hashes are still there as long as the user is logged in

- Kernel cat had to gain domain admin credentials to move **laterally**
- The credentials are **potentially** stored in the LSASS process
- But we can’t access them from usermode...

OBJECTIVES:

- [ ] INFILTRATE KERNEL
- [ ] ACCESSING LSASS MEMORY
- [ ] EXTRACT ADMIN HASHES

GOAL: MEET LADY CAT

KERNEL MITIGATIONS

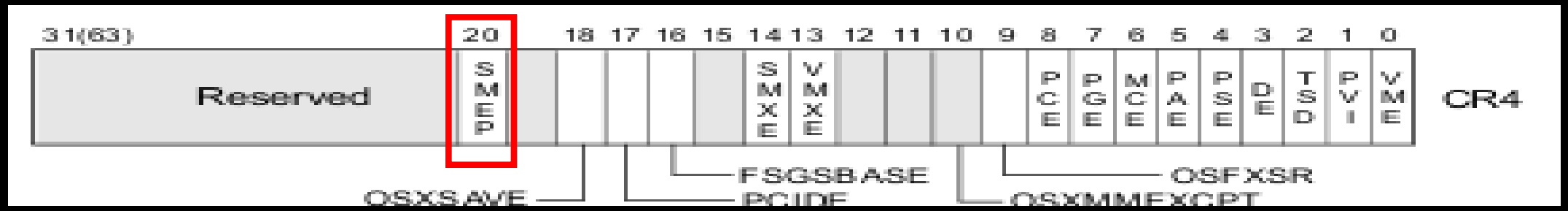
THINGS THAT KERNEL CAT NEEDS TO OVERCOME

- Data structure hardening
- KASLR
- KDEP
- SMEP

SMEP

SUPERVISOR MODE EXECUTION PROTECTION

- What?
  - Most LPEs requires running usermode shellcode
  - SMEP stops usermode code running from kernelmode
- How?
  - Modern OSs use virtual memory which is divided into pages (PTEs)
  - **PTE contains an owner bit (kernel/user)**
  - CPU enforces that kernel mode running kernel space pages - **Otherwise, TRAP!**
  - **OS enables SMEP feature by setting the 20th bit in CR4**



KNOWN TECHNIQUES TO BYPASS SMEP

- Disable 20th bit in CR4 using kernel ROP (ptsecurity)
- Crafting malicious kernel objects in RWX pools (j00ru)
- Patching Owner bit in the PXEs from user to kernel (MWR Info Sec)

BYPASS SMEP

A NEW TECHNIQUE

- Create a special kernel object
- User data (e.g. shellcode) is copied to kernel

→Data's PTE owner bit is now kernel → circumventing SMEP

Usermode API example:

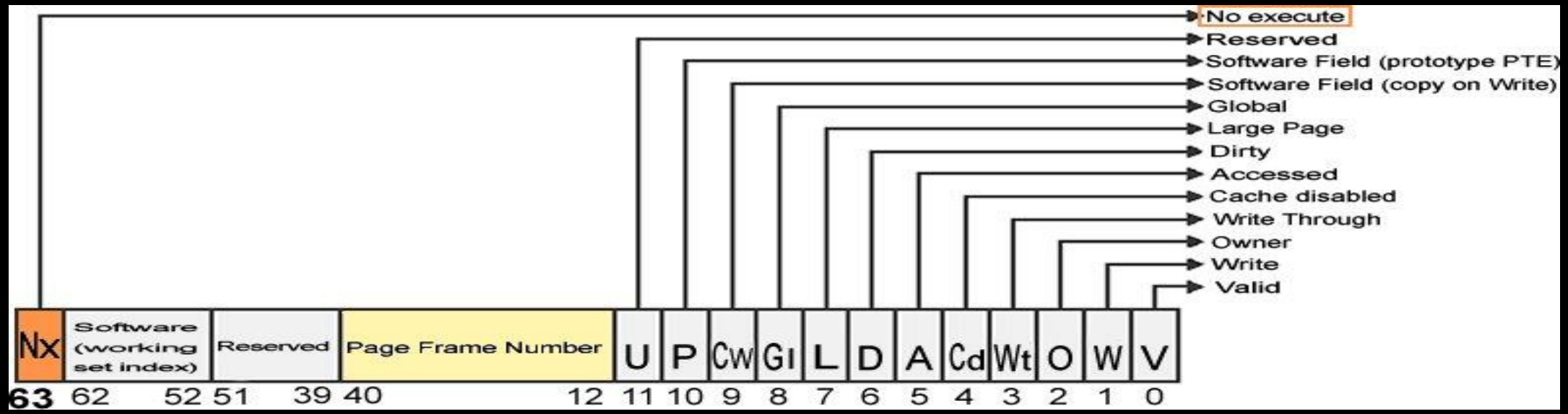
```
MENUITEMINFO item = {0};
item.dwTypeData = L"USER_DEFINED_CODE";
item.fType = MFT_STRING;
InsertMenuItem(hMenu, 0, 0, &item);
```

- However, kernel objects are allocated in NX pools (since Windows 8.1)

SMASH KERNEL DEP

1 BIT TO RULE THEM ALL

- Patching the PTE's NX bit



WHERE ART THOU PTE ?

- PTEs Base Address is located in a fixed memory address
  - Across all Windows versions
- For x86 bit: 0xC0000000
- For x64 bit: 0xFFFFF68000000000

→ KASLR is circumvented as well

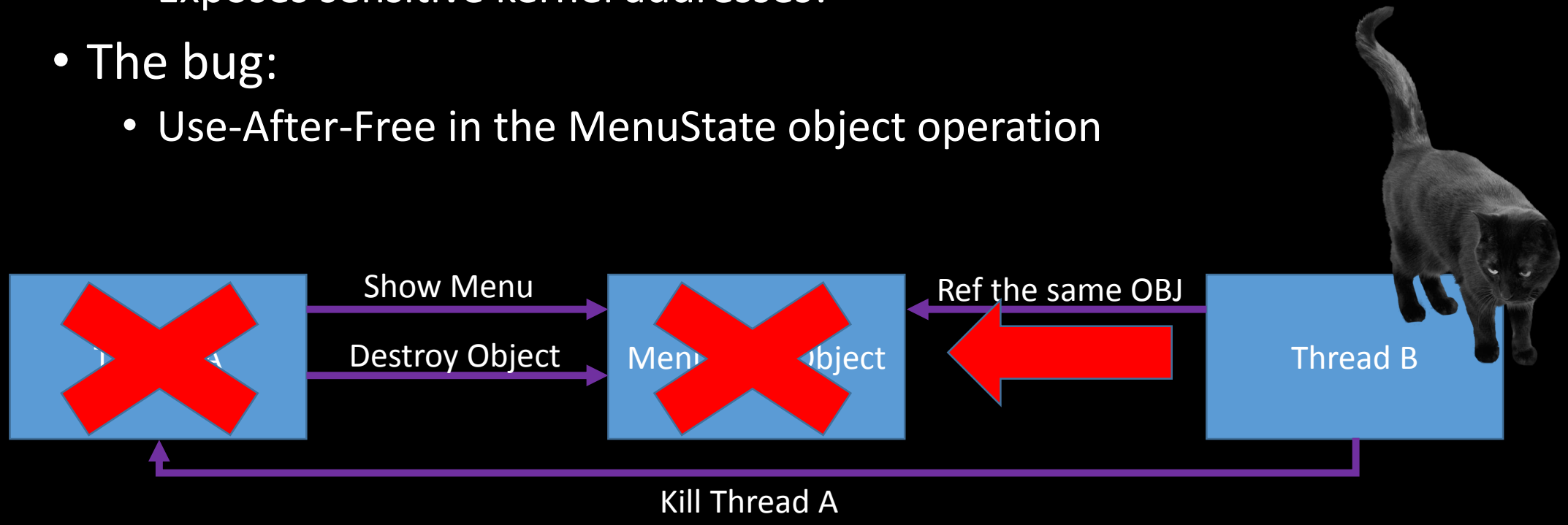
- Each virtual address' PTE is determined with a simple x86 formula:

$$PTE\_VA = PTE\_BASE + ((kOBJ\_VA >> 12) << 3)$$

ARMING A 1-DAY

CAUSE CATS DON'T WASTE GOOD 0-DAYS

- Win32k.sys:
  - kernel component responsible for all GUI in Windows
  - Maps lots of kernel objects into user space (for performance)
  - Exposes sensitive kernel addresses!
- The bug:
  - Use-After-Free in the MenuState object operation



DEMO - KERNEL INFILTRATION

- Bypass KASLR
- Bypass KDEP
- Bypass SMEP

OBJECTIVES:

- [x] INFILTRATE KERNEL
- [ ] ACCESSING LSASS MEMORY
- [ ] EXTRACT ADMIN HASHES

GOAL: MEET LADY CAT

Transferring control to the payload

- We need to get our shellcode executed (patched PTE) from ring0
- Use the write-anywhere primitive to patch a kernel callback pointer
- Trigger the callback in kernel mode

THE RING0 SHELLCODE

- Locating NTOSKRNL base address using sidt instruction
- Home made *GetProcAddress* code to find exported APIs
- Locating LSASS process
  - Walking the EPROCESS linked list (offset may break)
  - *ZwQuerySystemInformation*
  - Read LSASS PID from the registry
    - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\LsaPid
- Attach to the LSASS process with *KeStackAttachProcess*

DEMO – ACCESS LSASS FROM RING0

- Access LSASS memory from Ring0
- Printing its memory map

OBJECTIVES:

- [x] INFILTRATE KERNEL
- [x] ACCESSING LSASS MEMORY
- [ ] EXTRACT ADMIN HASHES

GOAL: MEET LADY CAT

EXTRACTING DOMAIN ADMIN HASHES

- Credentials hashes are stored in memory in a reversible way
- Encryption keys are stored in memory as well
- Searching LSASS memory for binary signatures
  - Point to the actual encrypted hashes and keys
- Decrypt based on the used algorithm (3DES/AES)

→ Output the decrypted domain admin hash

DEMO - EXTRACTING DOMAIN ADMIN HASHES

- Search LSASS memory for hash & keys
- Print decrypted domain hash

OBJECTIVES:

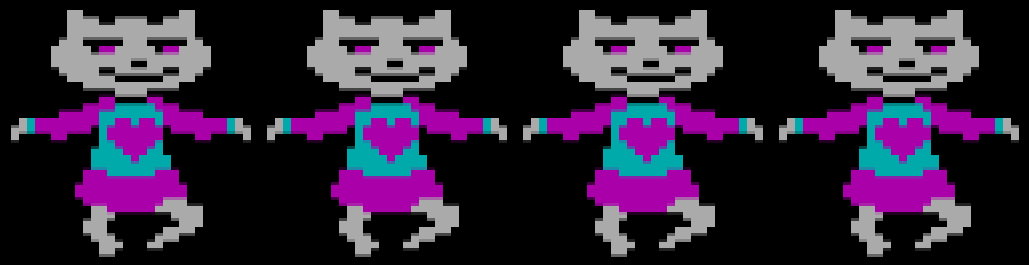
- ✓ INFILTRATE KERNEL
- ✓ DUMP LSASS MEMORY
- ✓ EXTRACT ADMIN HASHES

GOAL: MEET LADY CAT

SUMMARY

- Windows 8.1 security improved:
  - Protected Processes
  - New kernel mitigations
- Memory based attacks are stealth - Focus on live memory forensics
- Pass-The-* is here to stay
- Kernel access is a game over
  - Windows 10 will make it harder using vContainers - Upgrade now!
- Cats are awesome

THANK YOU



Gil Dabah

dabah@north-bit.com

@_arkon

http://north-bit.com

Tomer Teller

tomert@microsoft.com

@djteller