PROBING THE LIMITS OF VIRTUALIZED SOFTWARE PROTECTION

Joshua D. Cazalas, J.Todd McDonald, Todd R. Andel
University of South Alabama

Natalia Stakhanova
University of New Brunswick

Outline

- Research Overview
- Research Objectives
- Background and Related Research
- Methodology
- Results
- Future Work
- Questions?

Software Piracy

License Verification Code

```
mov  eax, ebx
push eax
add  ecx, eax
```

Original Code (x86 CPU)

Commercial Software

Illegal Copy

Illegal Copy

Illegal Copy

Illegal Copy

Stealing Intellectual Property

Proprietary Algorithm

```
mov  eax, ebx
push eax
add  ecx, eax
```

Original Code (x86 CPU)

Commercial Software

Secret Algorithm

Illegitimate Software

Software Watermarking

License Verification Code

```
mov  eax, ebx
push eax
add  ecx, eax
```

Original Code (x86 CPU)

Commercial Software

Code Virtualization

- Version 1
- Version 2
- Version 3
- Version 4

IP Protection

Proprietary Algorithm

```
mov  eax, ebx
push eax
add  ecx, eax
```

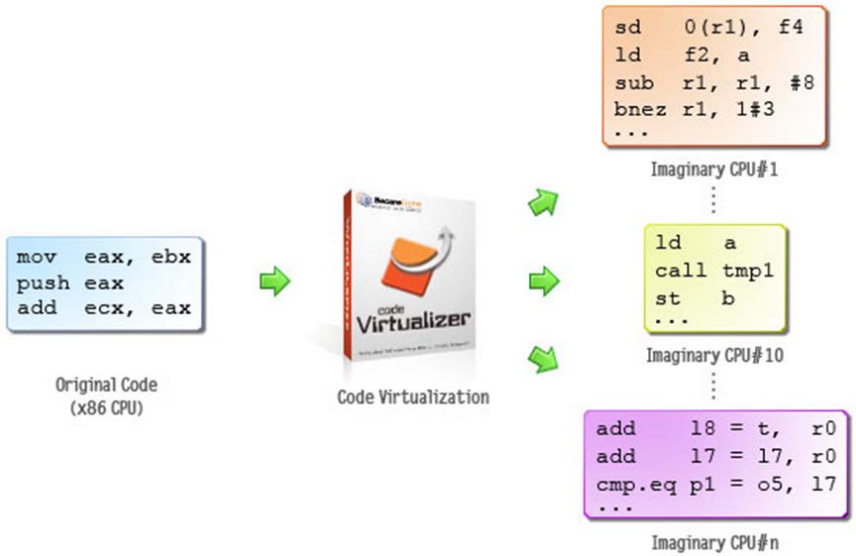
Original Code (x86 CPU)

Commercial Software

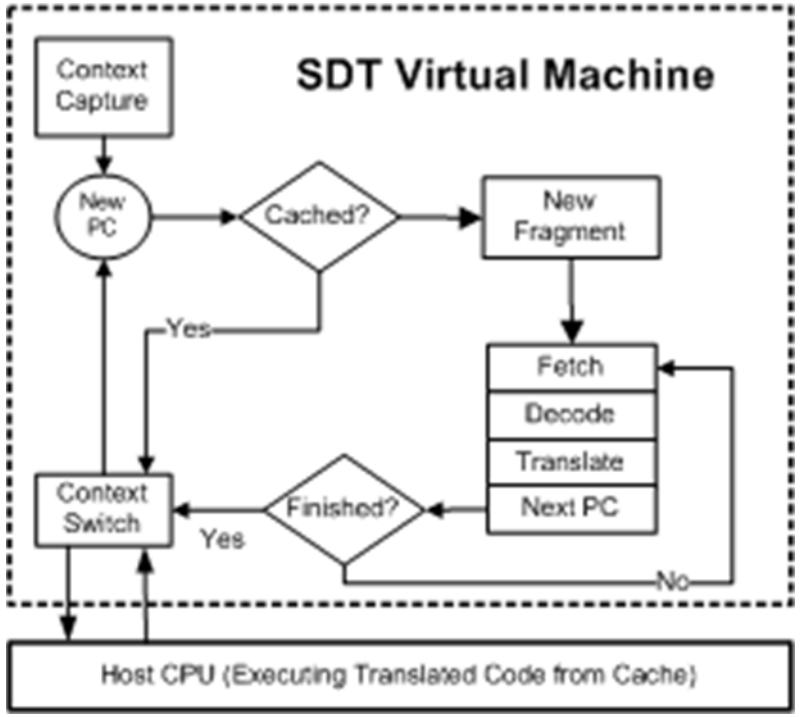
Code Virtualization

Secret Algorithm

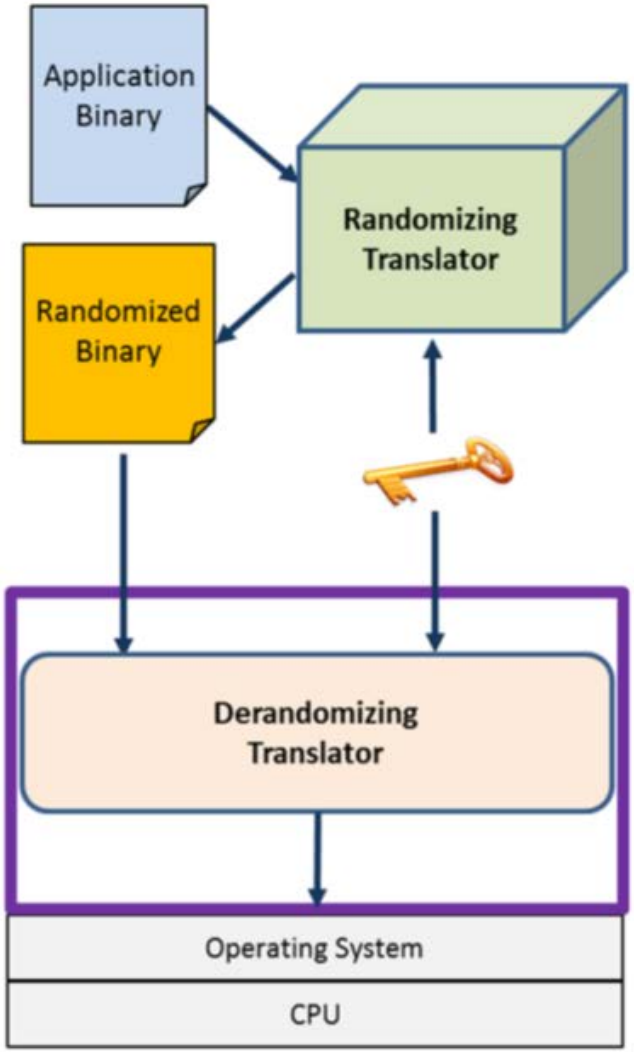
Virtualized Software



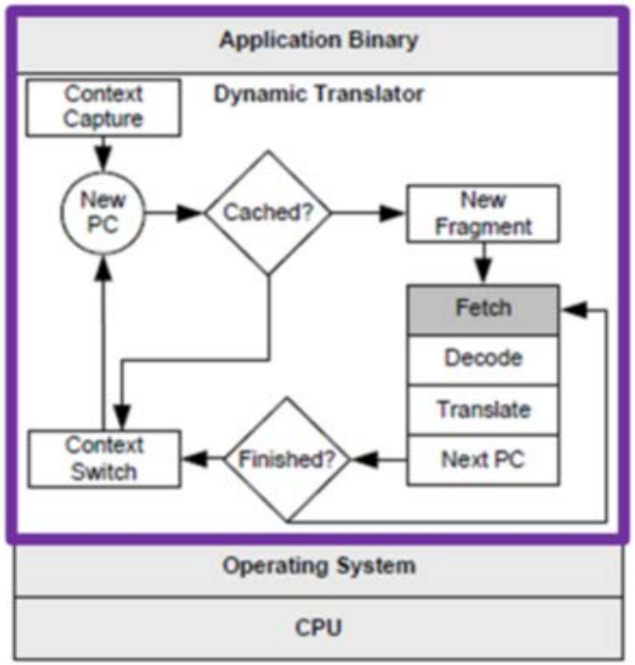
Virtualization(Static)



Virtualization(Dynamic)

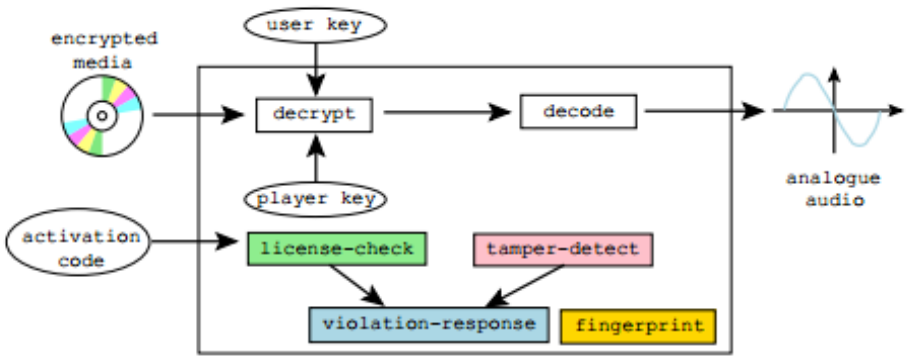


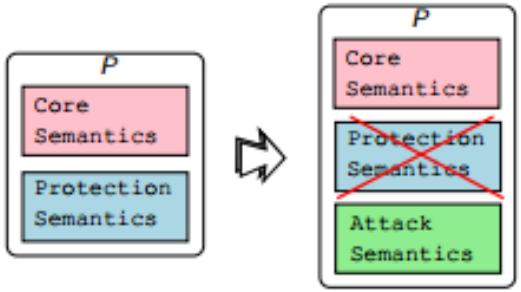
Static Translation



Dynamic Translation

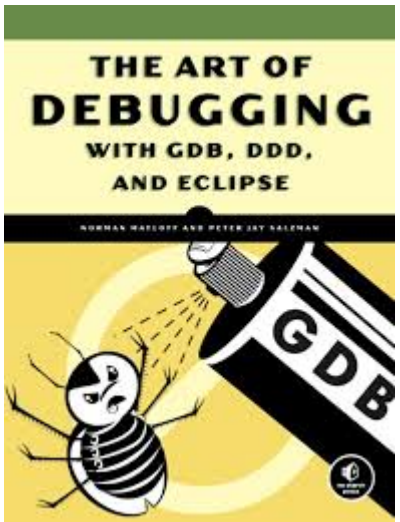
- General Purpose
  - Protect software algorithms from man at the end attacks
    - Code injection techniques
    - Reverse engineering
  - Force the attacker into a rewrite attack. (Collberg)





- General Purpose
  - Protect software algorithms from man at the end attacks

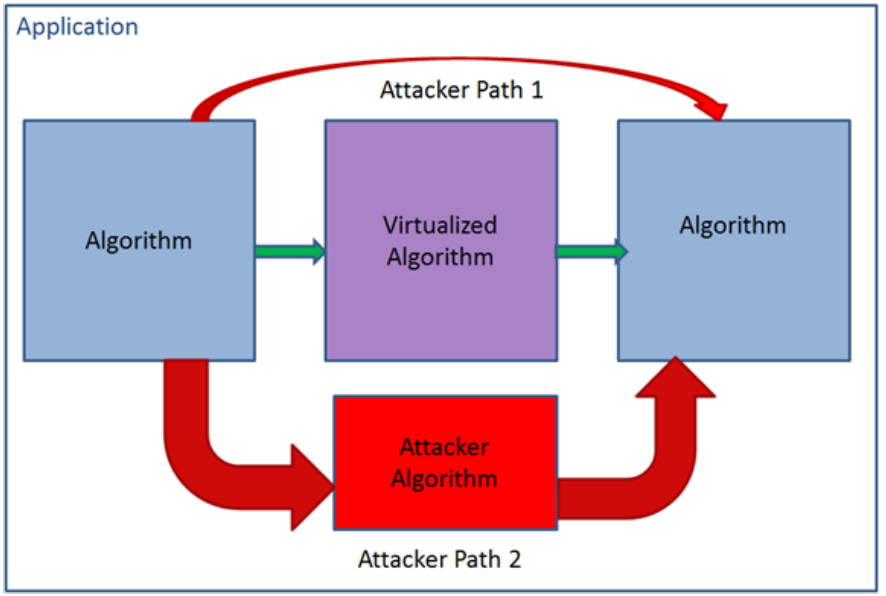
Static Analysis



Dynamic Analysis

- Explore existing virtualization techniques.
- Examine known subversion methods.
- Introduce an alternative subversion method(s).
- Suggest improvements to existing techniques to better protect legitimate use of virtualization in software protection.

- What is code injection?
  - The ultimate goal of a code injection attack is to change the control flow or flow of execution in a software system in such a way that anti-cloning resilience mechanism are subverted or intellectual property protections are nullified.
  - General case
    - Piracy
    - Minor software manipulation.



- Example:
  - When executing a piece of software "*IP*" (intellectual property) the following may be expected:

    Enter a Valid License Key:
  - With some output based on the users input:

    The license key is not valid. Please check with your vendor.
  - OR:

    The license key is valid.

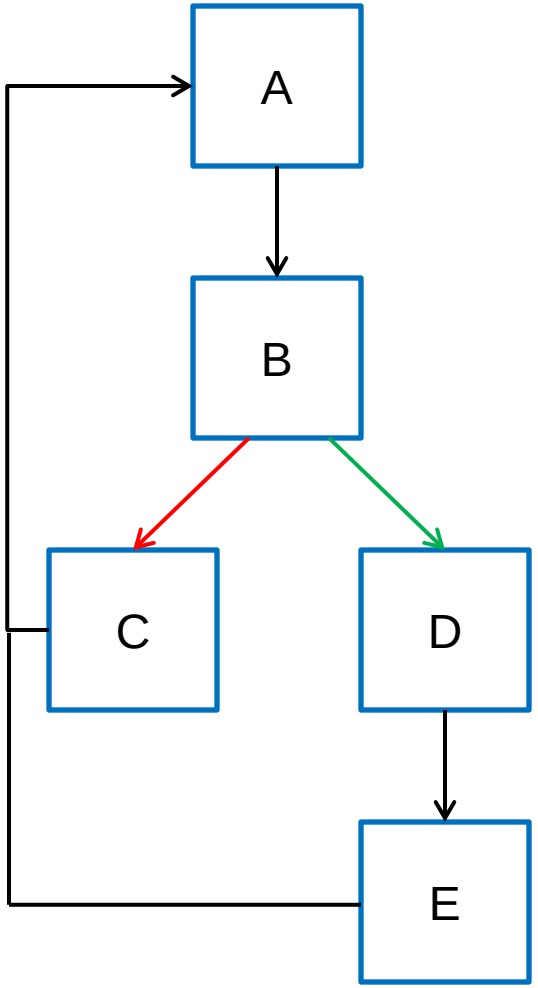
    You may execute the protected IP.

    Starting to execute protected IP.

    Finished executing protected IP.

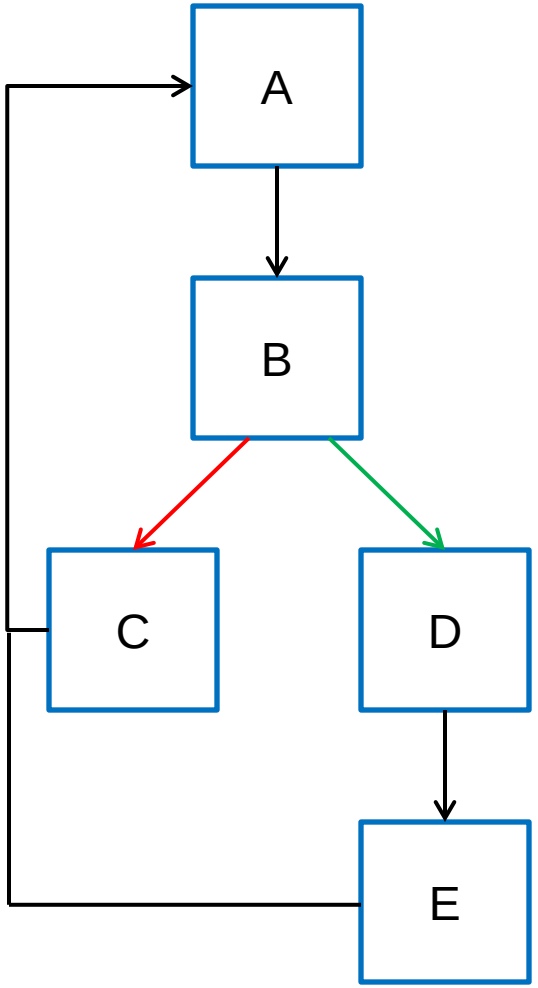
- Example:

```
Dump of assembler code for function checkLicense:
 0x08048361 <+0>: push %ebp
 0x08048362 <+1>: mov %esp,%ebp
 0x08048364 <+3>: sub $0x18,%esp
 0x08048367 <+6>: call 0x80482d8 <getLicense>
 0x0804836c <+11>: movl $0x80a7bfe,0x4(%esp)
 0x08048374 <+19>: mov %eax,(%esp)
 0x08048377 <+22>: call 0x8054b40
 0x0804837c <+27>: test %eax,%eax
 0x0804837e <+29>: jne 0x8048393
<checkLicense+50>
 0x08048380 <+31>: movl $0x80a7c0c,(%esp)
 0x08048387 <+38>: call 0x80492e0
 0x0804838c <+43>: call 0x8048250 <IP>
 0x08048391 <+48>: jmp 0x804839f
<checkLicense+62>
 0x08048393 <+50>: movl $0x80a7c48,(%esp)
 0x0804839a <+57>: call 0x80492e0
 0x0804839f <+62>: leave
 0x080483a0 <+63>: ret
End of assembler dump.
```



- Example:

```
Dump of assembler code for function checkLicense:
 0x08048361 <+0>: push %ebp
 0x08048362 <+1>: mov %esp,%ebp
 0x08048364 <+3>: sub $0x18,%esp
 0x08048367 <+6>: call 0x80482d8 <getLicense>
 0x0804836c <+11>: movl $0x80a7bfe,0x4(%esp)
 0x08048374 <+19>: mov %eax,(%esp)
 0x08048377 <+22>: call 0x8054b40
 0x0804837c <+27>: test %eax,%eax
 0x0804837e <+29>: jne 0x8048393 <checkLicense+50>
 0x08048380 <+31>: movl $0x80a7c0c,(%esp)
 0x08048387 <+38>: call 0x80492e0
 0x0804838c <+43>: call 0x8048250 <IP>
 0x08048391 <+48>: jmp 0x804839f <checkLicense+62>
 0x08048393 <+50>: movl $0x80a7c48,(%esp)
 0x0804839a <+57>: call 0x80492e0
 0x0804839f <+62>: leave
 0x080483a0 <+63>: ret
End of assembler dump.
```



- Example:

Using GDB we perform the following changes which modify the jne byte changing it to a NOP:

- set *(unsigned char*)0x804837e=0x90
- set *(unsigned char*)0x804837f=0x90

And we observe the following outputs after running the program:

- The license key is valid.

  You may execute the protected IP.

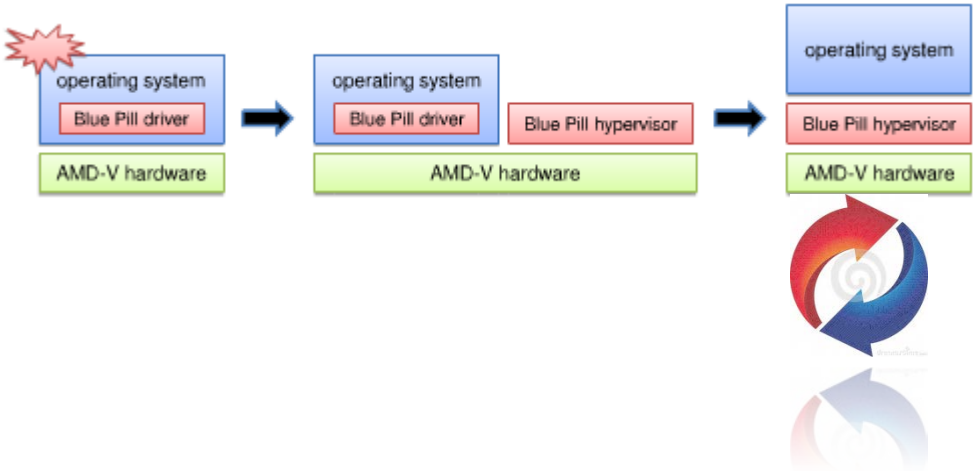
  Starting to execute protected IP.

  Finished executing protected IP.

```
Dump of assembler code for function checkLicense:
 0x08048361 <+0>: push %ebp
 0x08048362 <+1>: mov %esp,%ebp
 0x08048364 <+3>: sub $0x18,%esp
 0x08048367 <+6>: call 0x80482d8 <getLicense>
 0x0804836c <+11>: movl $0x80a7bfe,0x4(%esp)
 0x08048374 <+19>: mov %eax,(%esp)
 0x08048377 <+22>: call 0x8054b40
 0x0804837c <+27>: test %eax,%eax
 0x0804837e <+29>: jne 0x8048393
<checkLicense+50>
 0x08048380 <+31>: movl $0x80a7c0c,(%esp)
 0x08048387 <+38>: call 0x80492e0
 0x0804838c <+43>: call 0x8048250 <IP>
 0x08048391 <+48>: jmp 0x804839f
<checkLicense+62>
 0x08048393 <+50>: movl $0x80a7c48,(%esp)
 0x0804839a <+57>: call 0x80492e0
 0x0804839f <+62>: leave
 0x080483a0 <+63>: ret
End of assembler dump.
```

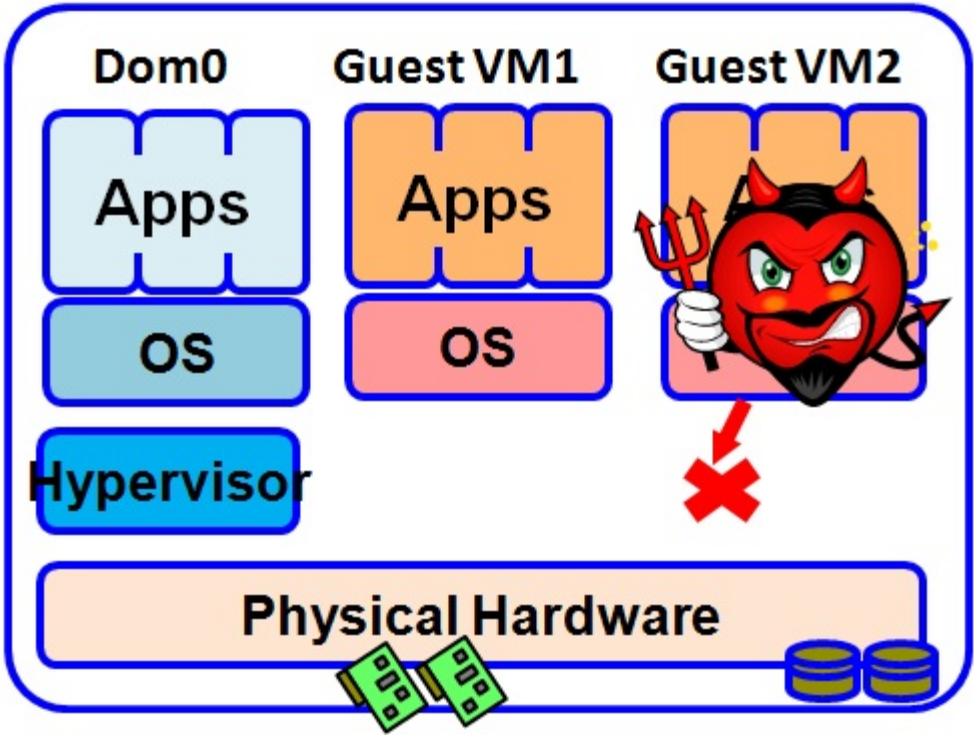
Can virtualization protect against this?

- RedPill/Blue Pill (Rutkowska)



- Cyclical nature of obfuscation and security in general.
- Lack of cohesion and coupling. (Scott 2003)
  - Ghosh/Hu/Davidson 2012

```
int swallow_redpill () {
  unsigned char m[2+4], rpill[] = "\x0f\x01\x0d\x00\x00\x00\x00\xc3";
  *((unsigned*)&rpill[3]) = (unsigned)m;
  ((void(*)())&rpill)();
  return (m[5]>0xd0) ? 1 : 0;
}
```



Two Prerequisites

1. The attacker must be able to locate the entry function (EP) of the protective PVM in PV .
2. The attacker must be aware of the guest application's instruction set architecture.

- Rolles

1. Reverse engineer the virtual machine.
2. Detect locations at which control flow enters the virtualization obfuscator.
3. Develop a procedure for producing a disassembler, given a protected executable.
4. Disassemble the byte code and convert it into intermediate code.
5. Apply compiler optimizations to the IR.
6. Generate x86 code.

- Rolles
  - Requires a highly skilled attacker
    - Large amounts of reverse engineering
    - Architecture specific
    - Costly and time consuming
  - Highly Effective
  - Still usually cheaper than rewrite attacks

1. Create a standardized target program for protection.
2. Choose a representative protection technique (Forms of PVM).
3. For each representative technique perform MATE attacks.
4. Provide measurements. (Overhead or proof of concept)

2. Static PVM

- Static Analysis
  - Algorithm only virtualizers have trivially locatable entry points.
  - Full code virtualizers are persistent and require additional analysis. May agree with Rolle's findings.
- Dynamic analysis
  - Dynamic injection could be precise if performed manually.
  - Shotgun approaches are effective.
  - Brute force methods can be effective.

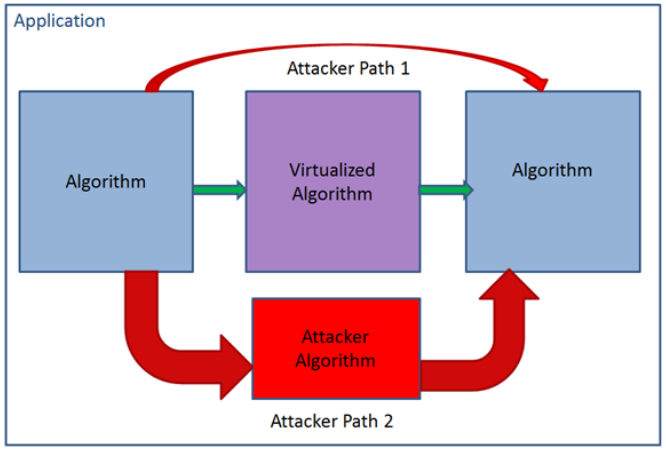
2. Dynamic PVM

- Static Analysis
  - Highly obfuscated
  - Static analysis alone is almost completely ineffective
- Dynamic Analysis
  - Entry point location is negligible
  - Swap functions create patterns
  - Hash table's in STRATA create multiple attack vectors.

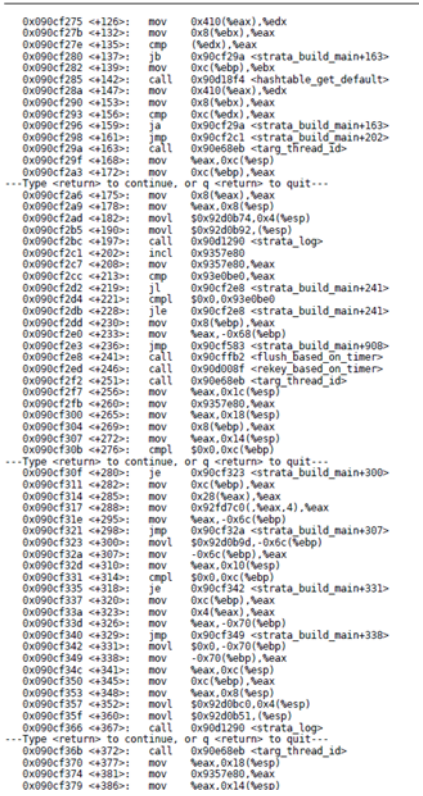
3. Perform Mate Attacks

- Static Virtualizers
  - Vulnerable to shotgun and brute force as predicted
  - Vulnerable to Rolle's Method
- Dynamic Virtualizers
  - Highly effective against all automated injections.
  - Vulnerable to Ghosh method

3. Perform Mate focusing on PVM



- Static Virtualizers
  - Fully resistant
    - Code is stored in a persistent virtual state
    - Reliant on the interpreter
    - High Coupling
- Dynamic Virtualizers
  - Trivial subversion given the entry point
  - Multiple known methods for entry points detection




```
=> 0x090cf203 <+12>:    call   0x90d18f4 <hashtable_get_default>
   0x090cf208 <+17>:    movl   $0x1,0x8(%eax)
   0x090cf20f <+24>:    mov    0xc(%ebp),%eax
   0x090cf212 <+27>:    mov    %eax,0x4(%esp)
   0x090cf216 <+31>:    mov    0x8(%ebp),%eax
   0x090cf219 <+34>:    mov    %eax,(%esp)
   0x090cf21c <+37>:    call   0x90cf034 <strata_enter_builder>
   0x090cf221 <+42>:    call   0x90d18f4 <hashtable_get_default>
   0x090cf226 <+47>:    cmpl   $0x0,0x424(%eax)
   0x090cf22d <+54>:    jne    0x90cf236 <strata_build_main+63>
```

```
   0x090cf296 <+159>:   ja     0x90cf29a <strata_build_main+163>
   0x090cf298 <+161>:   jmp    0x90cf2c1 <strata_build_main+202>
   0x090cf29a <+163>:   call   0x90e68eb <targ_thread_id>
   0x090cf29f <+168>:   mov    %eax,0xc(%esp)
   0x090cf2a3 <+172>:   mov    0xc(%ebp),%eax
  -Type <return> to continue, or q <return> to quit---
   0x090cf2a6 <+175>:   mov    0x8(%eax),%eax
   0x090cf2a9 <+178>:   mov    %eax,0x8(%esp)
   0x090cf2ad <+182>:   movl   $0x92d0b74,0x4(%esp)
   0x090cf2b5 <+190>:   movl   $0x92d0b92,(%esp)
   0x090cf2bc <+197>:   call   0x90d1290 <strata_log>
   0x090cf2c1 <+202>:   incl   0x9357e80
   0x090cf2c7 <+208>:   mov    0x9357e80,%eax
   0x090cf2cc <+213>:   cmp    0x93e0be0,%eax
   0x090cf2d2 <+219>:   jl     0x90cf2e8 <strata_build_main+241>
   0x090cf2d4 <+221>:   cmpl   $0x0,0x93e0be0
   0x090cf2db <+228>:   jle    0x90cf2e8 <strata_build_main+241>
   0x090cf2dd <+230>:   mov    0x8(%ebp),%eax
   0x090cf2e0 <+233>:   mov    %eax,-0x68(%ebp)
   0x090cf2e3 <+236>:   jmp    0x90cf583 <strata_build_main+908>
   0x090cf2e8 <+241>:   call   0x90cffb2 <flush_based_on_timer>
   0x090cf2ed <+246>:   call   0x90d008f <rekey_based_on_timer>
```

- Dynamic Virtualizers
  - Recognizable Entry Pattern
  - 73.3% (11/15 trial runs) success rate with shotgun approach after the pattern was deduced.

- Dynamic Virtualizers

  - Given the success rate of the deduced pattern an automated algorithm could be implemented directly as a script.

  - Small changes to STRATA would render that script ineffective and a representative or modeled approach may be more relevant.

4. Overhead Analysis

- Static Virtualizers
  - Conclusion that the Rolle's research is accurate.
- Dynamic Virtualizers
  - Linear increase in security
  - Size/time: 1x – 3x

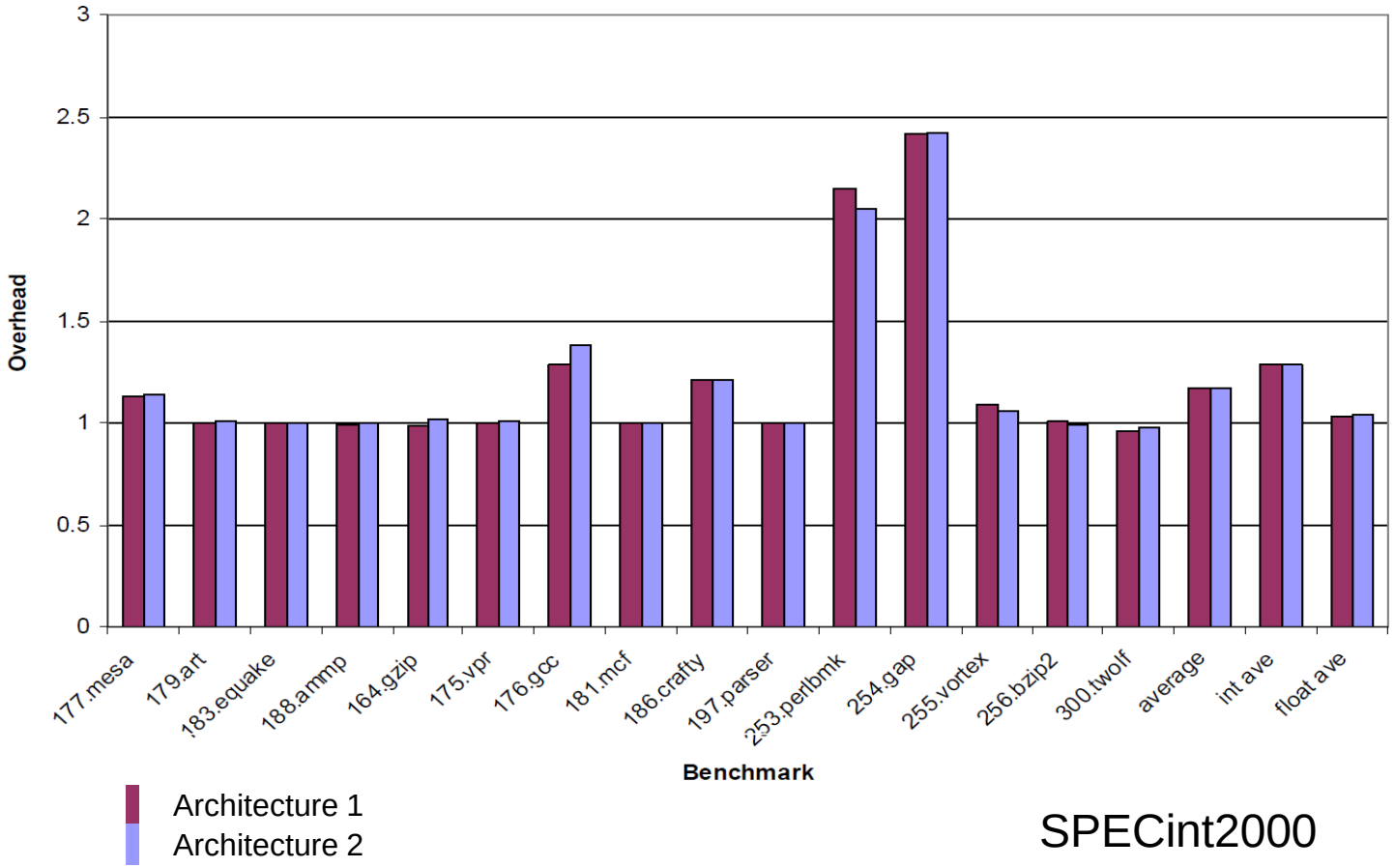
- Dynamic Virtualizers

	BIND				Apache			
	Native	**SDT-only**	**SDT-based ISR**	**SDT-based ISR Expansion**	**Native**	**SDT-only**	**SDT-based ISR**	**SDT-based ISR Expansion**
Disk image	1811	1872	2731	1.51x	916	987	1617	1.77x
text	1786	1838	2690	1.51x	875	939	1566	1.79x
data	23	28	32	1.39x	34	39	43	1.26x
bss	13	32	41	3.15x	166	186	194	1.17x

Table 3: Resident set size overhead (Megabytes).

Zone File Size (BIND) or Web Page Size (Apache)	BIND				Apache			
	Native	**SDT-only**	**SDT-based ISR**	**Expansion**	**Native**	**SDT-only**	**SDT-based ISR**	**Expansion**
1K	1.6	5.7	7.0	4.38x	N/A	N/A	N/A	N/A
10K	3.5	7.5	8.9	2.54x	0.7	3.0	3.6	5.14x
100K	22.0	26.0	27.4	1.24x	0.7	3.0	3.6	5.14x
1000K	N/A	N/A	N/A	N/A	0.7	3.0	3.6	5.14x

- Dynamic Virtualizers



SPECint2000

- Static virtualizer packages vary from extremely effective to almost completely ineffective.
  - Full code virtualization (high coupling)
  - Algorithm virtualization (low coupling)
- Dynamic virtualizers suffer from an additional low coupling problem.

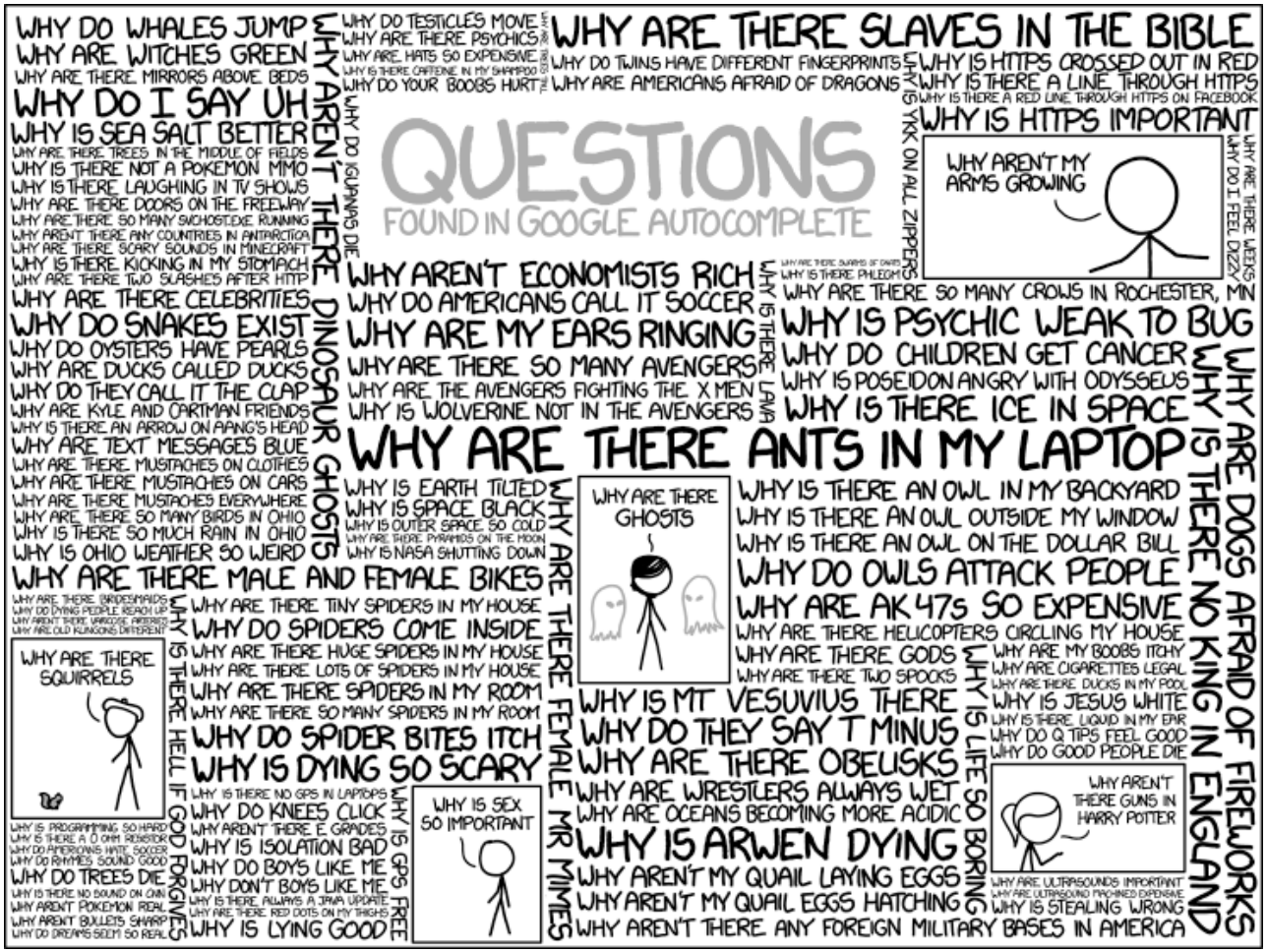
- Effective static virtualizers reduce the number of automated attack but remain vulnerable to shotgun and brute force methods.
  - Algorithm only virtualizer have negligible benefit given their overhead costs.
- They also require the Rolles' method at a minimum for persistent effective injection.
  - High cost manual process

- Dynamic virtualizers can fully protect the code against automated injection.
  - Can’t protect themselves
  - Low coupling
  - Very high overhead costs
- Low cost attacks can bypass.
  - Entry point injection
  - Ghosh Method
- Interestingly they make the Rolle’s method even more costly.

- Static/Dynamic Hybrids
  - Static virtualizers suffer from automated brute force style injections
  - Dynamic virtualizers suffer from low coupling
  - Each protects against the others flaw
  - Research into the costs and benefits of a hybrid approach could be performed

- Virtual Dependency
  - Design the software to run reliant on the hypervisor
  - Decreases overhead/Software engineering issue
  - Ghosh subversion approach still viable
- Multi-Process
  - Armadillo
- Hardware Integration
  - Adds additional coupling
  - Arc Injection Protection
  - Cryptographically secure block signing

- ✓Explore existing virtualization techniques.
  - -Static Virtualization and Software Dynamic Translation
- ✓Examine known subversion methods.
  - -Ghosh and Rolles
- ✓Introduce an alternative subversion method.
  - -Manual Context Switch Injection
- ✓Suggest improvements to existing techniques to better protect legitimate use of virtualization in software protection.
  - -Hybrids, Dependencies, or Hardware Integration


QUESTIONS
FOUND IN GOOGLE AUTOCOMPLETE
WHY ARE THERE ANTS IN MY LAPTOP
WHY ARE THERE SLAVES IN THE BIBLE
WHY DO WHALES JUMP
WHY ARE WITCHES GREEN
WHY ARE THERE MIRRORS ABOVE BEDS
WHY DO I SAY UH
WHY IS SEA SALT BETTER
WHY ARE THERE TREES IN THE MIDDLE OF FIELDS
WHY IS THERE NOT A POKEMON MMO
WHY IS THERE LAUGHING IN TV SHOWS
WHY ARE THERE DOORS ON THE FREEWAY
WHY ARE THERE SO MANY SVCHOST.EXE RUNNING
WHY AREN'T THERE ANY COUNTRIES IN ANTARCTICA
WHY ARE THERE SCARY SOUNDS IN MINECRAFT
WHY IS THERE KICKING IN MY STOMACH
WHY ARE THERE TWO SLASHES AFTER HTTP
WHY ARE THERE CELEBRITIES
WHY DO SNAKES EXIST
WHY DO OYSTERS HAVE PEARLS
WHY ARE DUCKS CALLED DUCKS
WHY DO THEY CALL IT THE CLAP
WHY ARE KYLE AND CARTMAN FRIENDS
WHY IS THERE AN ARROW ON AANG'S HEAD
WHY ARE TEXT MESSAGES BLUE
WHY ARE THERE MUSTACHES ON CLOTHES
WHY ARE THERE MUSTACHES ON CARS
WHY ARE THERE MUSTACHES EVERYWHERE
WHY ARE THERE SO MANY BIRDS IN OHIO
WHY IS THERE SO MUCH RAIN IN OHIO
WHY IS OHIO WEATHER SO WEIRD
WHY ARE THERE MALE AND FEMALE BIKES
WHY ARE THERE BRIDESMAIDS
WHY DO DYING PEOPLE REACH UP
WHY AREN'T THERE VARICOSE AMBERES
WHY ARE OLD KLINGONS DIFFERENT
WHY ARE THERE SQUIRRELS
WHY IS PROGRAMMING SO HARD
WHY IS THERE A 0 OHM RESISTOR
WHY DO AMERICANS HATE SOCCER
WHY DO RHYMES SOUND GOOD
WHY DO TREES DIE
WHY IS THERE NO SOUND ON ONN
WHY AREN'T POKEMON REAL
WHY AREN'T BULLETS SHARP
WHY DO DREAMS SEEM SO REAL
WHY DO DINOSAUR GHOSTS
WHY AREN'T THERE
WHY DO IGUANAS DIE
WHY DO TESTICLES MOVE
WHY ARE THERE PSYCHICS
WHY ARE HATS SO EXPENSIVE
WHY IS THERE CAFFEINE IN MY SHAMPOO
WHY DO YOUR BOOBS HURT
WHY DO TWINS HAVE DIFFERENT FINGERPRINTS
WHY ARE AMERICANS AFRAID OF DRAGONS
WHY IS HTTPS CROSSED OUT IN RED
WHY IS THERE A LINE THROUGH HTTPS
WHY IS THERE A RED LINE THROUGH HTTPS ON FACEBOOK
WHY IS HTTPS IMPORTANT
WHY AREN'T MY ARMS GROWING
WHY ARE THERE WEEKS
WHY DO I FEEL DIZZY
WHY ARE THERE SWANS AT PARKS
WHY IS THERE PHLEGM
WHY ARE THERE SO MANY CROWS IN ROCHESTER, MN
WHY AREN'T ECONOMISTS RICH
WHY DO AMERICANS CALL IT SOCCER
WHY ARE MY EARS RINGING
WHY ARE THERE SO MANY AVENGERS
WHY ARE THE AVENGERS FIGHTING THE X MEN
WHY IS WOLVERINE NOT IN THE AVENGERS
WHY IS THERE LAVA
WHY IS PSYCHIC WEAK TO BUG
WHY DO CHILDREN GET CANCER
WHY IS POSEIDON ANGRY WITH ODYSSEUS
WHY IS THERE ICE IN SPACE
WHY IS EARTH TILTED
WHY IS SPACE BLACK
WHY IS OUTER SPACE SO COLD
WHY ARE THERE PYRAMIDS ON THE MOON
WHY IS NASA SHUTTING DOWN
WHY ARE THERE GHOSTS
WHY ARE THERE TINY SPIDERS IN MY HOUSE
WHY DO SPIDERS COME INSIDE
WHY ARE THERE HUGE SPIDERS IN MY HOUSE
WHY ARE THERE LOTS OF SPIDERS IN MY HOUSE
WHY ARE THERE SPIDERS IN MY ROOM
WHY ARE THERE SO MANY SPIDERS IN MY ROOM
WHY DO SPIDER BITES ITCH
WHY IS DYING SO SCARY
WHY IS THERE NO GPS IN LAPTOPS
WHY DO KNEES CLICK
WHY AREN'T THERE E GRADES
WHY IS ISOLATION BAD
WHY DO BOYS LIKE ME
WHY DON'T BOYS LIKE ME
WHY IS THERE ALWAYS A JAVA UPDATE
WHY ARE THERE RED DOTS ON MY THIGHS
WHY IS LYING GOOD
WHY IS SEX SO IMPORTANT
WHY IS GPS FREE
WHY ARE THERE FEMALE MR MIMES
WHY IS MT VESUVIUS THERE
WHY DO THEY SAY T MINUS
WHY ARE THERE OBELISKS
WHY ARE WRESTLERS ALWAYS WET
WHY ARE OCEANS BECOMING MORE ACIDIC
WHY IS ARWEN DYING
WHY AREN'T MY QUAIL LAYING EGGS
WHY AREN'T MY QUAIL EGGS HATCHING
WHY AREN'T THERE ANY FOREIGN MILITARY BASES IN AMERICA
WHY IS THERE AN OWL IN MY BACKYARD
WHY IS THERE AN OWL OUTSIDE MY WINDOW
WHY IS THERE AN OWL ON THE DOLLAR BILL
WHY DO OWLS ATTACK PEOPLE
WHY ARE AK 47s SO EXPENSIVE
WHY ARE THERE HELICOPTERS CIRCLING MY HOUSE
WHY ARE THERE GODS
WHY ARE THERE TWO SPOOKS
WHY ARE MY BOOBS ITCHY
WHY ARE CIGARETTES LEGAL
WHY ARE THERE DUCKS IN MY POOL
WHY IS JESUS WHITE
WHY IS THERE LIQUID IN MY EAR
WHY DO Q TIPS FEEL GOOD
WHY DO GOOD PEOPLE DIE
WHY AREN'T THERE GUNS IN HARRY POTTER
WHY ARE ULTRASOUNDS IMPORTANT
WHY ARE ULTRASOUND MACHINES EXPENSIVE
WHY IS STEALING WRONG
WHY IS THERE NO KING IN ENGLAND
WHY ARE DOGS AFRAID OF FIREWORKS
WHY IS LIFE SO BORING
WHY IS THERE HELL IF GOD FORGIVES
WHY IS GPS FREE
WHY DO YKK ON ALL ZIPPERS