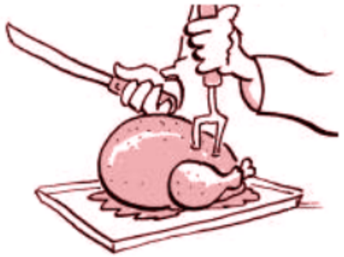
THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

GW SCHOOL OF ENGINEERING & APPLIED SCIENCE

# DamGate: Dynamic Adaptive Multi-feature Gating in Program Binaries

Yurong Chen, Tian Lan, Guru Venkataramani



ACM CCS FEAST 2017

Office of Naval Research
Science & Technology

# De-bloat: Make software “slimmer” by removing unused features

- Guoqing Xu, Nick Mitchell, Matthew Arnold, Atanas Rountev, and Gary Sevitsky. "Software bloat analysis: finding, removing, and preventing performance problems in modern large-scale object-oriented applications."
- Guoqing Xu, Nick Mitchell, Matthew Arnold, Atanas Rountev, Edith Schonberg, and Gary Sevitsky. "Scalable runtime bloat detection using abstract dynamic slicing."
- Yufei Jiang, Dinghao Wu, and Peng Liu. "JRed: Program Customization and Bloatware Mitigation Based on Static Analysis."
- Yufei Jiang, Can Zhang, Dinghao Wu, and Peng Liu. "Feature-Based Software Customization: Preliminary Analysis, Formalization, and Methods.”

Static:

Remove unused code

Dynamic:

Remove inefficient runtime behavior

# What’s more?

- Undesired interactions among different features
- Varying user requirements

# Goal of Design

## Ideas

- Be compatible with de-bloating
- Customize binaries after de-bloating
- Prevent undesired interactions among features
- Instrument binary to add checker functions
- Enable dynamic reconfiguration of feature profile
- Multiple features are kept in binaries and managed by config file

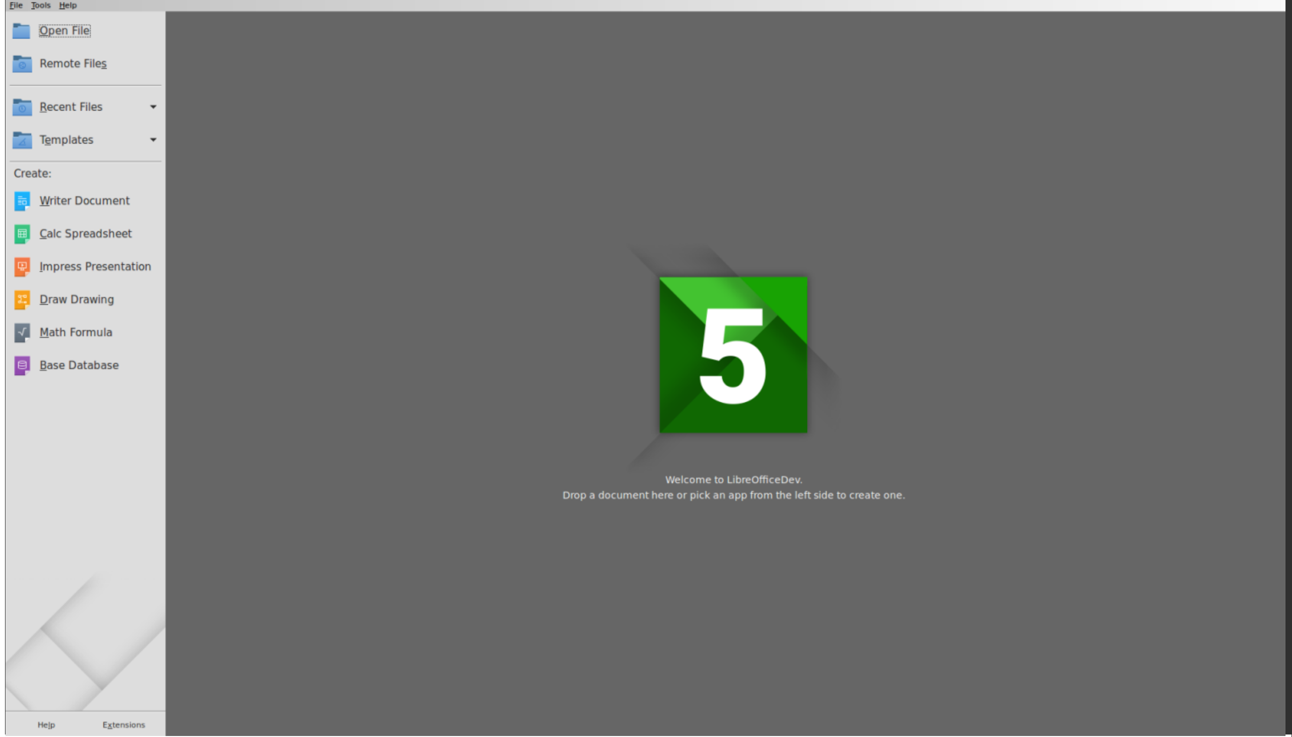
**DamGate**

# Definitions

- Each feature, denoted by $F^i$, is defined as a set of functions that can perform certain tasks independent of other functions outside this set, e.g., $F^i = \{ f_1^i , f_2^i , \dots, f_n^i \}$.

- Seed function $f_s^i \in F^i$, is used to identify the feature. Seed functions are representative functions of the features they belong to. (we assume features can be identified by seed functions: how to get the seed functions in practice;)

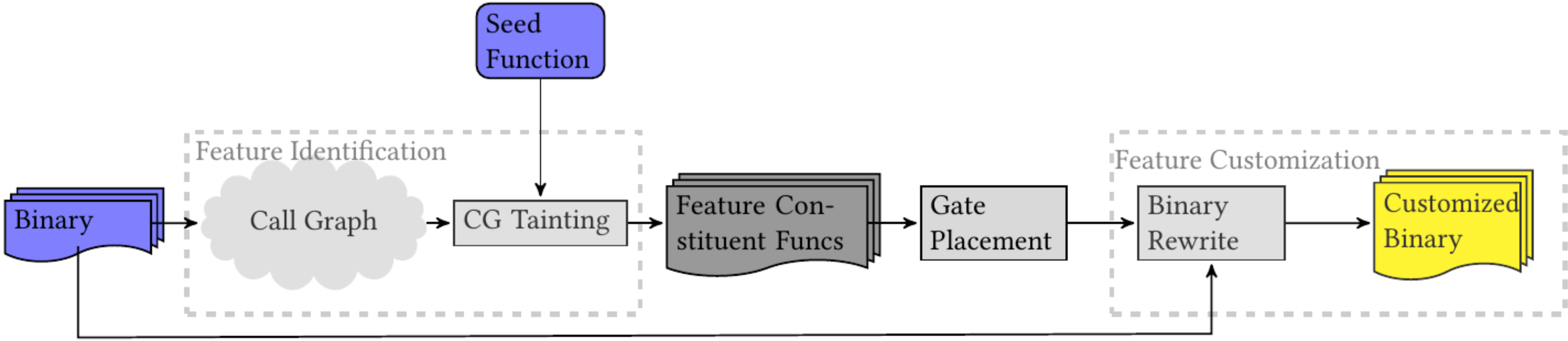
- Gate: a function checker that verify if the target function is allowed according to current protection policy.



Welcome to LibreOfficeDev.
Drop a document here or pick an app from the left side to create one.

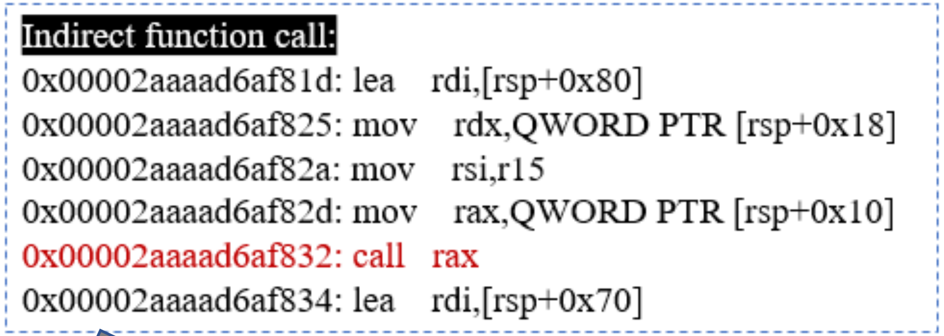
| Feature Name | Seed Function |
|---|---|
| Save file | SfxObjectShell::SaveTo_Impl |
| Print | SfxViewShell::ExecPrint |
| Insert image | SwView::InsertGraphic |

## Examples
(from Libre-office)



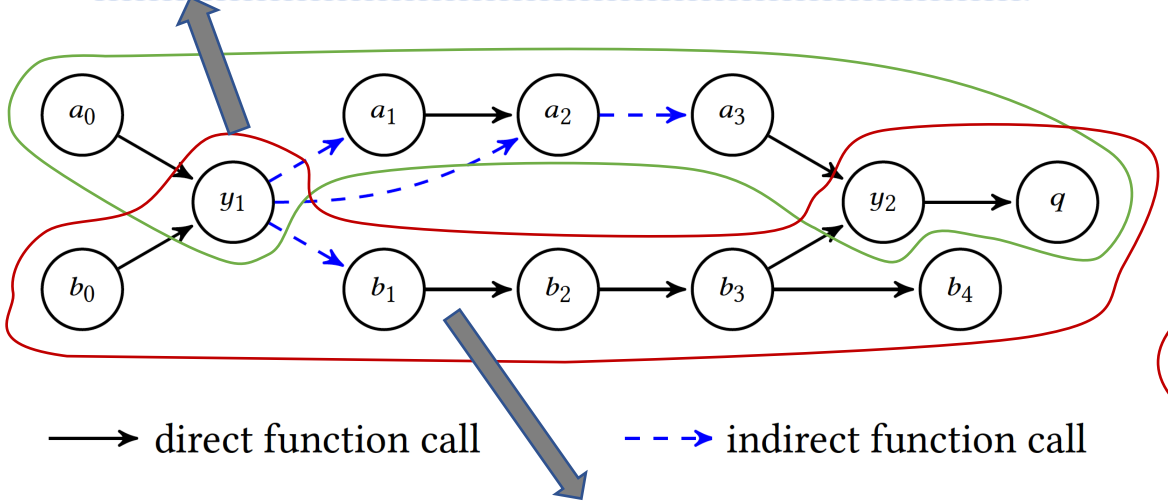
# System Overview

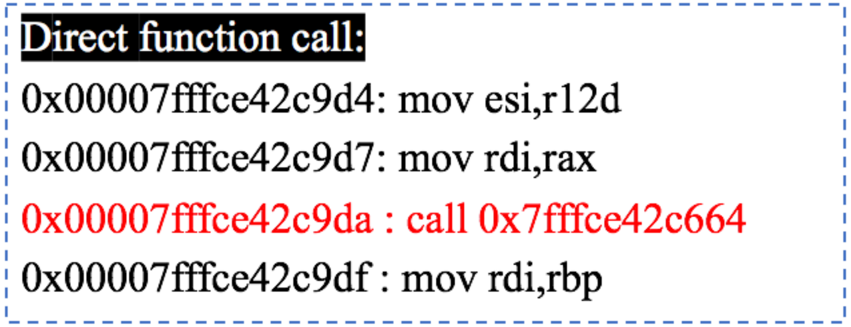
# Example: LibreOffice



**Indirect function call:**

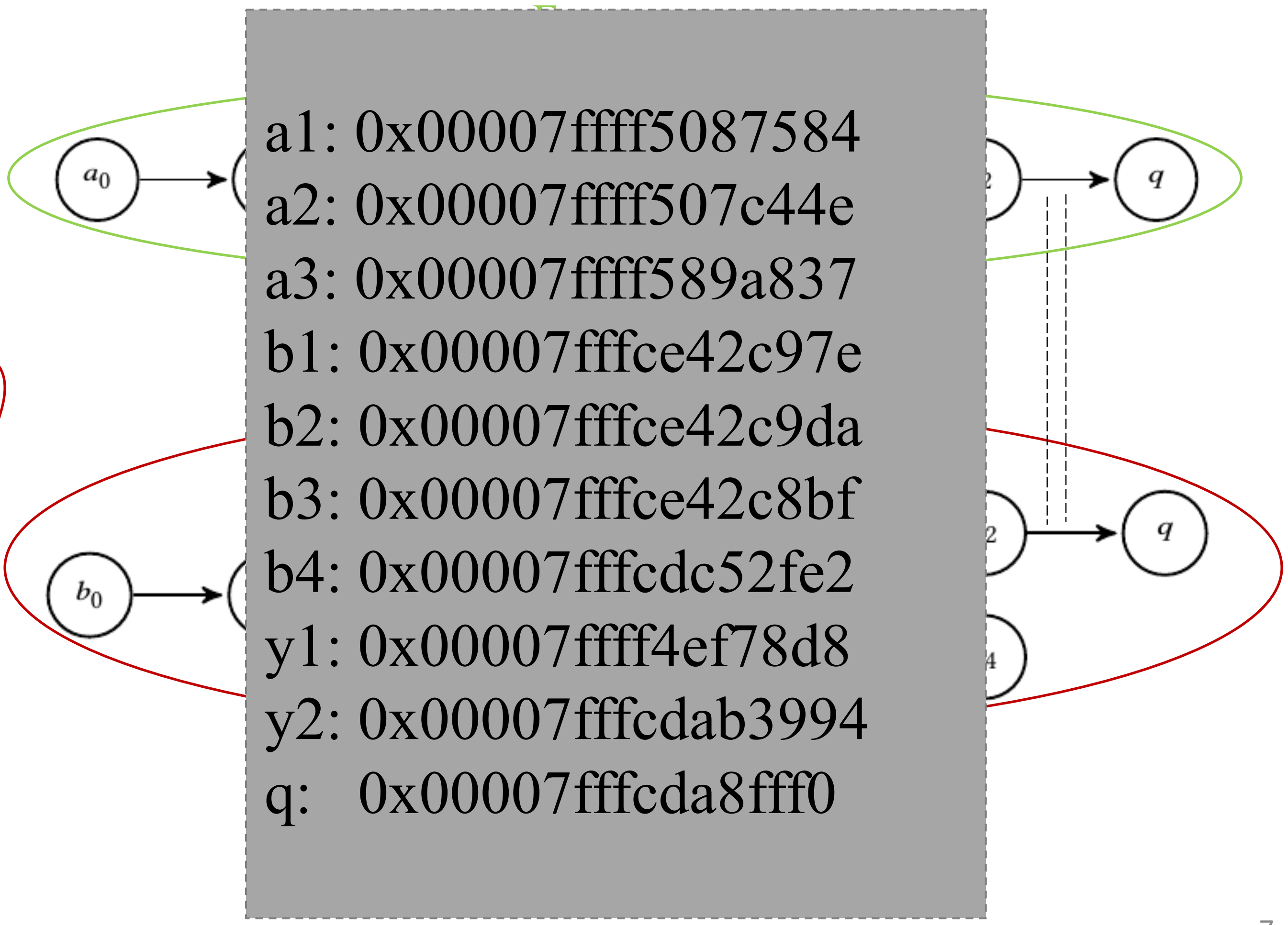
```
0x00002aaaad6af81d: lea    rdi,[rsp+0x80]
0x00002aaaad6af825: mov    rdx,QWORD PTR [rsp+0x18]
0x00002aaaad6af82a: mov    rsi,r15
0x00002aaaad6af82d: mov    rax,QWORD PTR [rsp+0x10]
0x00002aaaad6af832: call   rax
0x00002aaaad6af834: lea    rdi,[rsp+0x70]
```





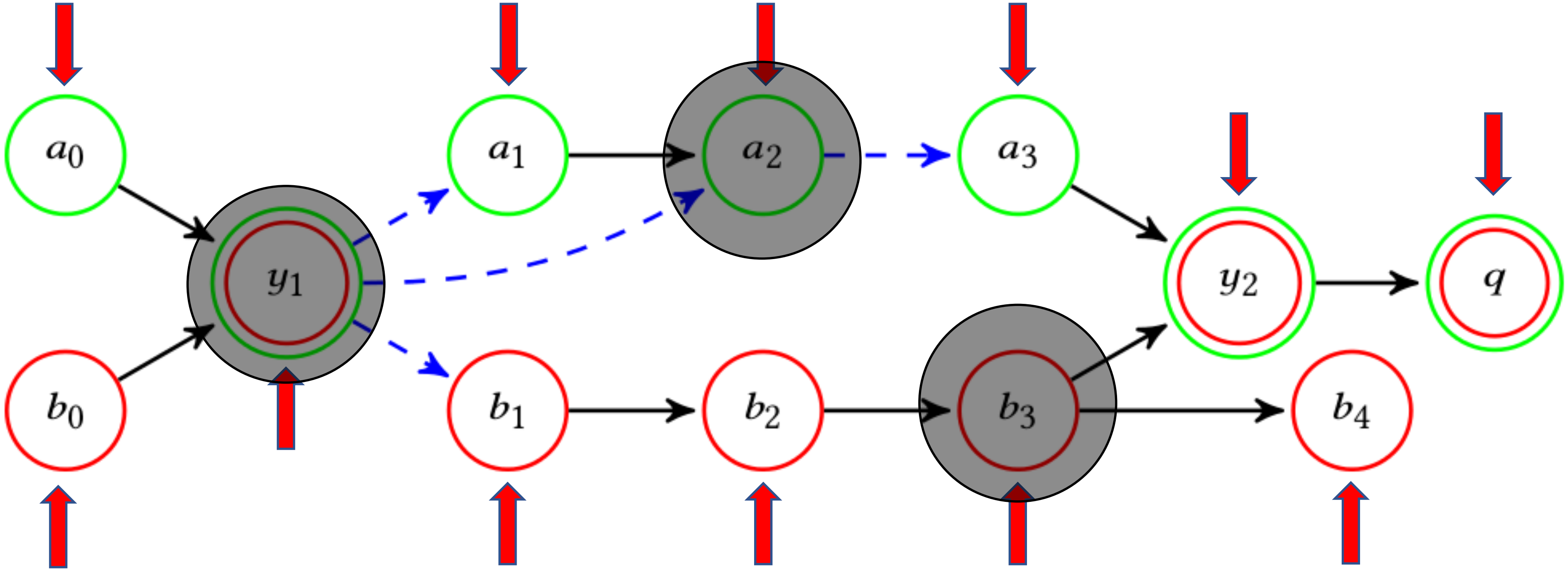
**Direct function call:**

```
0x00007fffce42c9d4: mov esi,r12d
0x00007fffce42c9d7: mov rdi,rax
0x00007fffce42c9da : call 0x7fffce42c664
0x00007fffce42c9df : mov rdi,rbp
```



| | |
|---|---|
| a1: | 0x00007ffff5087584 |
| a2: | 0x00007ffff507c44e |
| a3: | 0x00007ffff589a837 |
| b1: | 0x00007fffce42c97e |
| b2: | 0x00007fffce42c9da |
| b3: | 0x00007fffce42c8bf |
| b4: | 0x00007fffcdc52fe2 |
| y1: | 0x00007ffff4ef78d8 |
| y2: | 0x00007fffcdab3994 |
| q: | 0x00007fffcda8fff0 |

# Example



# Feature Identification

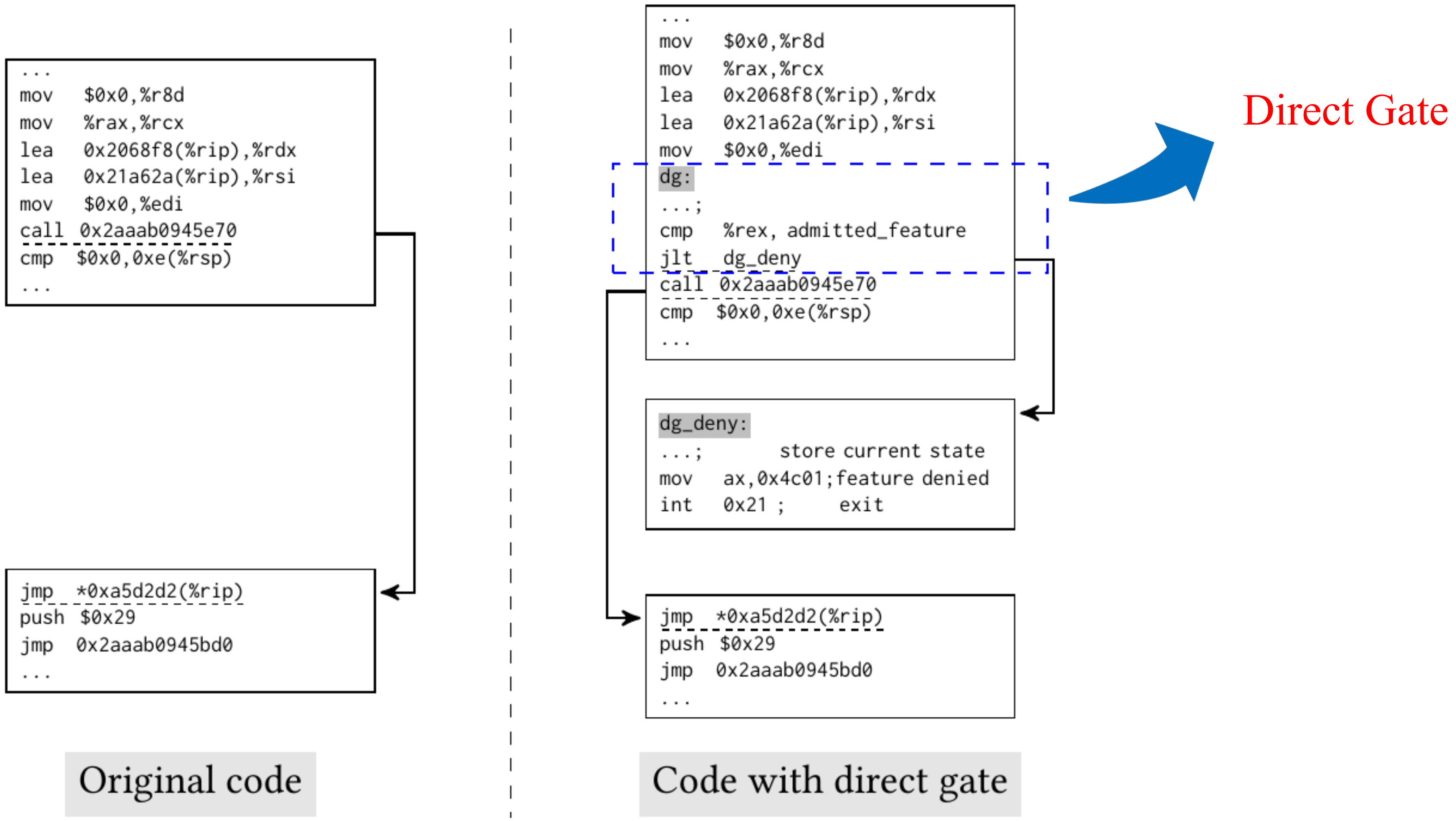
- Call Graph (CG) generation from binary:
  - Static tool: CodeSurfer (from GrammaTech)
  - Dynamic tool: Pin (from Intel)
- CG Tainting: CG + seed functions
- Mark function calls as direct or indirect ones in the CG

# Feature Customization

- Direct Gate : check feature
- Indirect Gate: check control transfer + feature
  - Martín Abadi, Mihai Budiu, Ulfar Erlingsson, and Jay Ligatti. "Control-flow integrity."
  - Chao Zhang, Tao Wei, Zhaofeng Chen, Lei Duan, Laszlo Szekeres, Stephen McCamant, Dawn Song, and Wei Zou. "Practical control flow integrity and randomization for binary executables.”
  - Ben Niu, and Gang Tan. "Modular control-flow integrity."

- Binary rewriting: Dyninst (from University of Wisconsin-Madison and University of Maryland)

# Feature Customization: Direct Gate



```
...
mov   $0x0,%r8d
mov   %rax,%rcx
lea   0x2068f8(%rip),%rdx
lea   0x21a62a(%rip),%rsi
mov   $0x0,%edi
call  0x2aaab0945e70
cmp  $0x0,0xe(%rsp)
...
```

```
jmp  *0xa5d2d2(%rip)
push  $0x29
jmp  0x2aaab0945bd0
...
```

Original code

```
...
mov   $0x0,%r8d
mov   %rax,%rcx
lea   0x2068f8(%rip),%rdx
lea   0x21a62a(%rip),%rsi
mov   $0x0,%edi
dg:
...;
cmp   %rex, admitted_feature
jlt   dg_deny
call  0x2aaab0945e70
cmp  $0x0,0xe(%rsp)
...
```

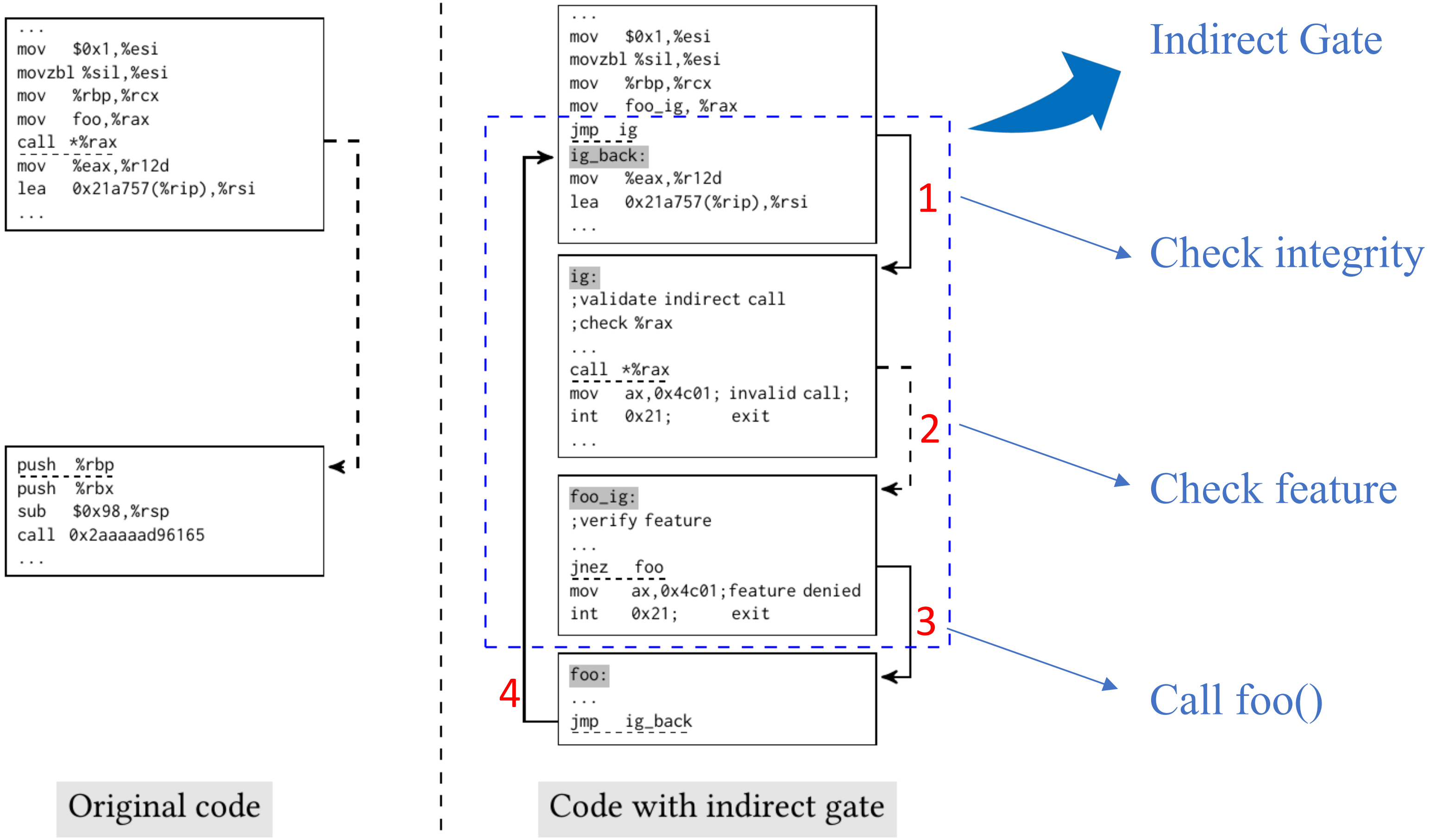
```
dg_deny:
...;        store current state
mov   ax,0x4c01;feature denied
int   0x21 ;      exit
```

```
jmp  *0xa5d2d2(%rip)
push  $0x29
jmp  0x2aaab0945bd0
...
```

Code with direct gate

Direct Gate

# Feature Customization: Indirect Gate



```
...
mov   $0x1,%esi
movzbl %sil,%esi
mov   %rbp,%rcx
mov   foo,%rax
call  *%rax
mov   %eax,%r12d
lea   0x21a757(%rip),%rsi
...
```

```
push  %rbp
push  %rbx
sub   $0x98,%rsp
call  0x2aaaaad96165
...
```

Original code

```
...
mov   $0x1,%esi
movzbl %sil,%esi
mov   %rbp,%rcx
mov   foo_ig, %rax
jmp   ig
ig_back:
mov   %eax,%r12d
lea   0x21a757(%rip),%rsi
...
```

```
ig:
;validate indirect call
;check %rax
...
call  *%rax
mov   ax,0x4c01; invalid call;
int   0x21;       exit
...
```

```
foo_ig:
;verify feature
...
jnez  foo
mov    ax,0x4c01;feature denied
int    0x21;      exit
```

```
foo:
...
jmp   ig_back
```

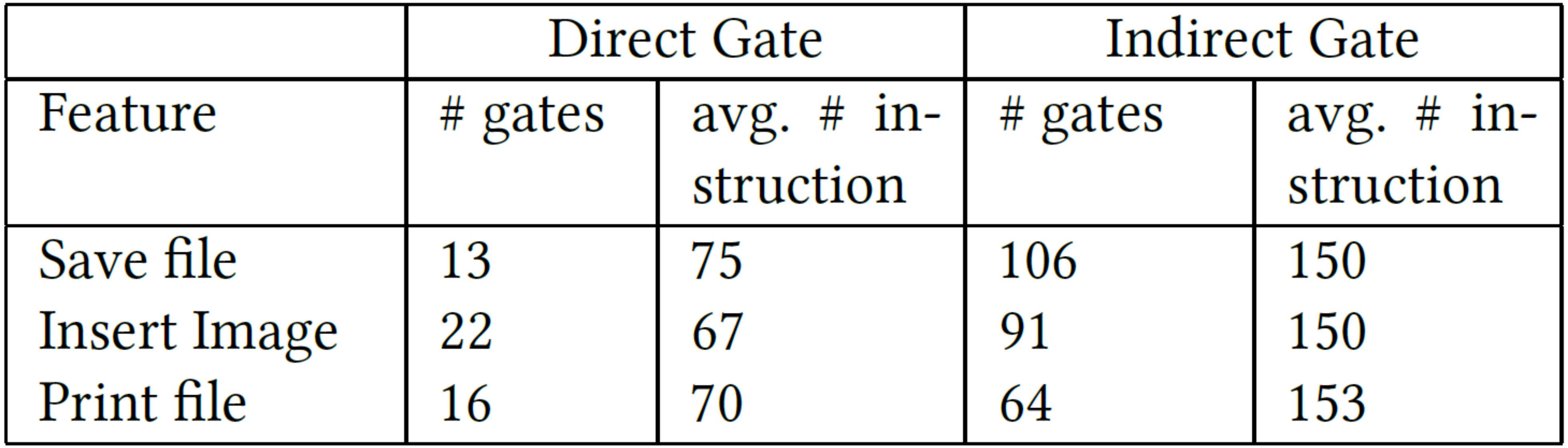
Code with indirect gate

Indirect Gate

Check integrity

Check feature

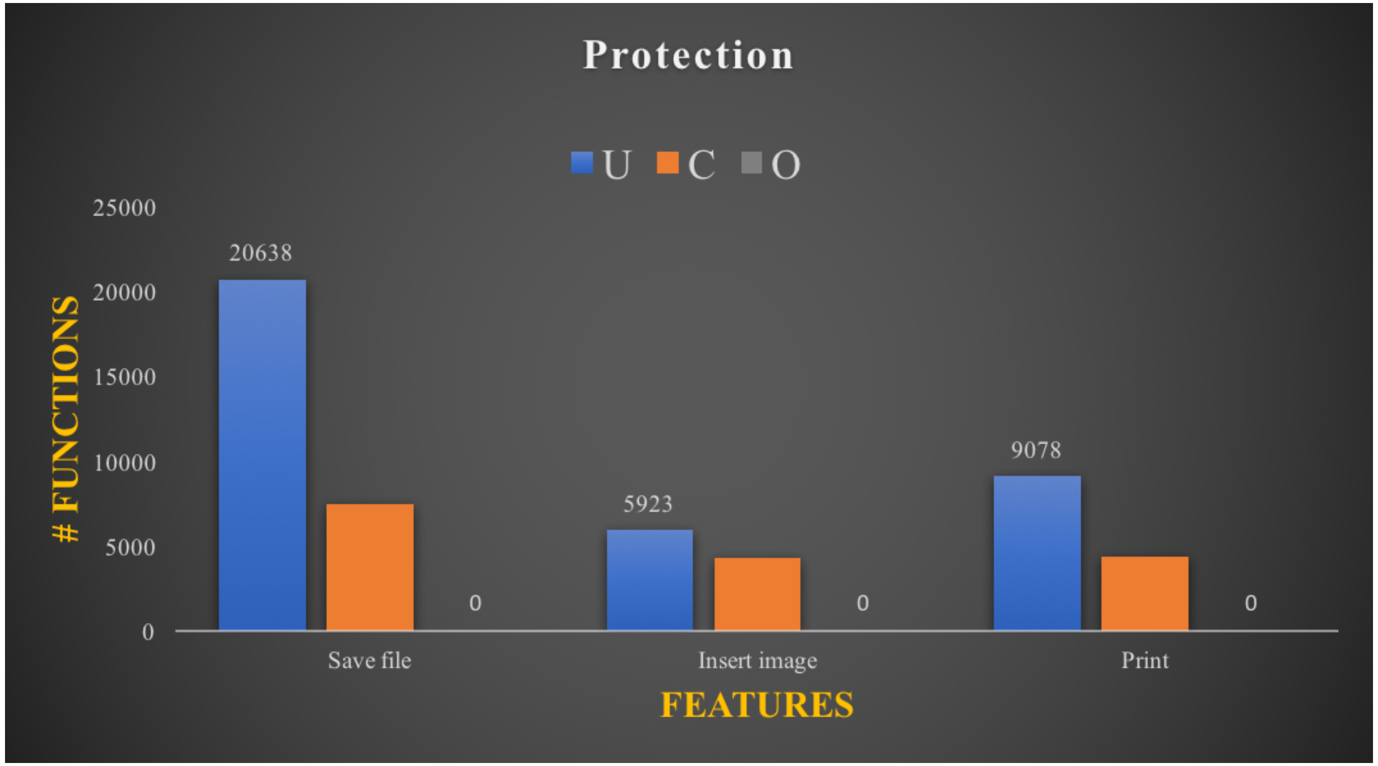
Call foo()



|  | Direct Gate |  | Indirect Gate |  |
|---|---|---|---|---|
| Feature | # gates | avg. # instruction | # gates | avg. # instruction |
| Save file | 13 | 75 | 106 | 150 |
| Insert Image | 22 | 67 | 91 | 150 |
| Print file | 16 | 70 | 64 | 153 |

# Evaluation

# LibreOffice



U: unique functions that only belong to current feature
C: common functions shared by current feature and other features
O: functions that don not belong to current feature but are still accessed

# Evaluation

# LibreOffice

# Summary & Future work

- Tools:
  - Feature identification: CodeSurfer, Pin
  - Feature customization: Dyninst

- Gating policy;

- Open source, automated framework

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

Office of Naval Research

ONR

Science & Technology

N00014-17-1-2786 & N00014-15-1-2210

# Thank you!