Special Topics in Security
ECE 5968

Engin Kirda
ek@ccs.neu.edu

Northeastern University

Admin News and Stuff

- Apologies for the correction delay
  - I’m correcting myself and I’ve been swamped
  - Should be done this weekend – high on my todo list

News from the field

- Lots of interesting things happening (all the time)

- One interesting news item: Google introducing "Advanced Protection Accounts"
  - Use of heavy two factor authentication
  - Account credential stealing is very popular, and two factor (without the phone) is very effective

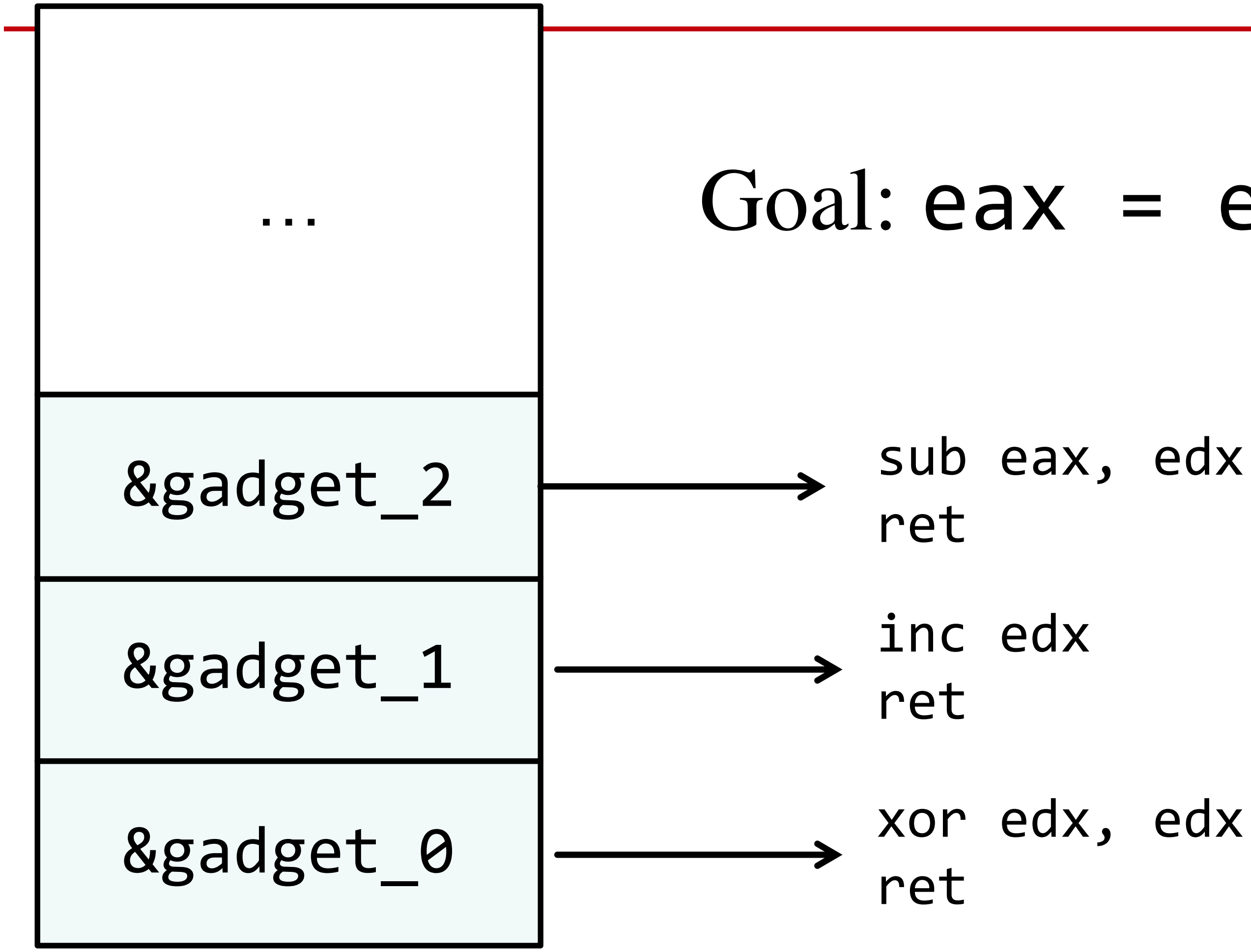
Return-Oriented Programming (ROP) Attacks

Return-Oriented Programming

- Return-oriented programming (ROP) extends return-into-libc
  - Introduced by Shacham in 2007
  - Shown to be Turing complete (for libc)!
  - But, in practice is used to disable memory protection
- Instead of reusing functions, ROP reuses *gadgets*
  - Gadgets are small sequences of instructions ending in a return
  - Each gadget performs some small update to the program state
  - Execution becomes a chain of returns to gadgets

Gadgets

Goal: `eax = eax - 1`



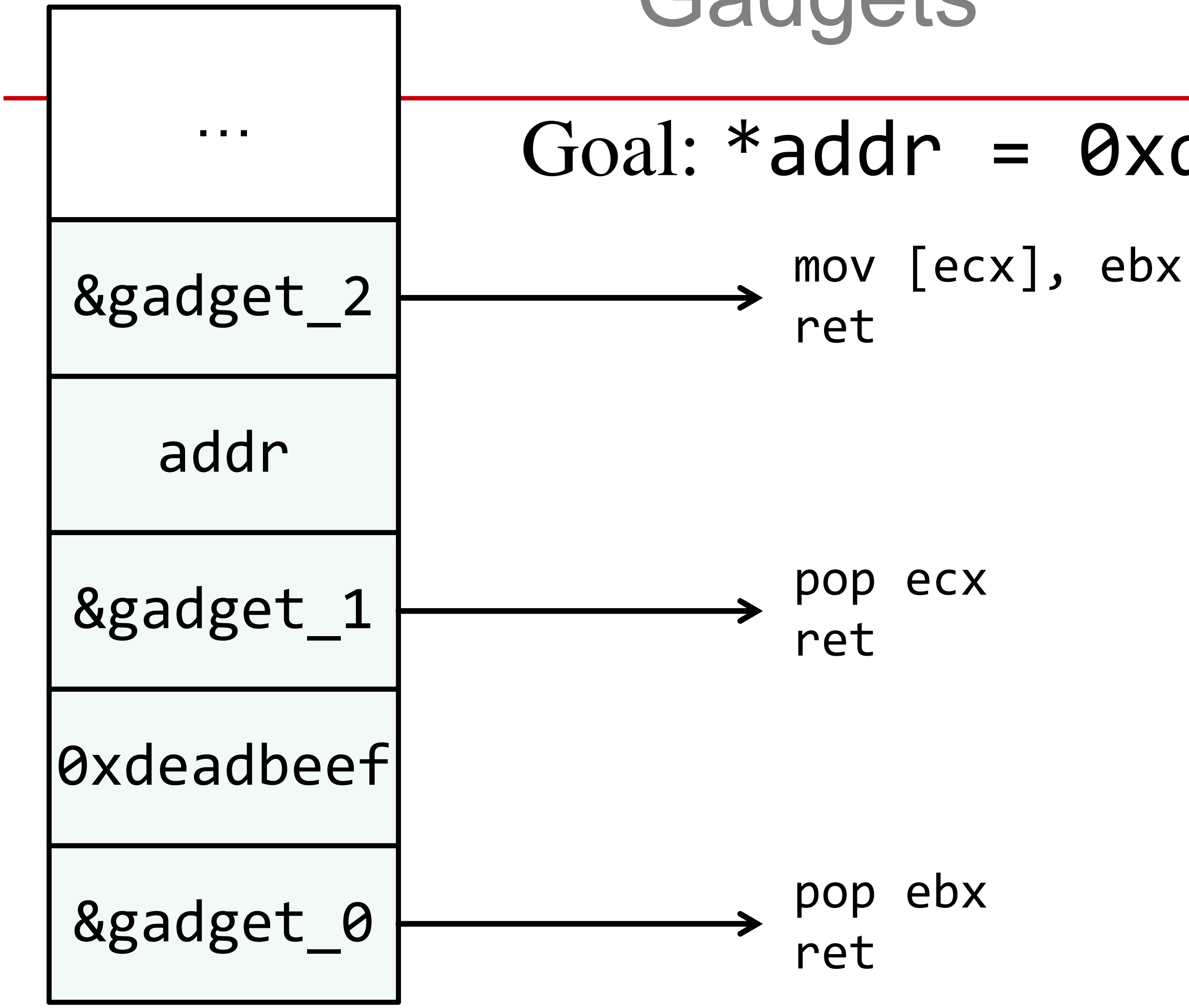
```
sub eax, edx
ret
```

```
inc edx
ret
```

```
xor edx, edx
ret
```

Gadgets

Goal: `*addr = 0xdeadbeef`



```
mov [ecx], ebx
ret
```

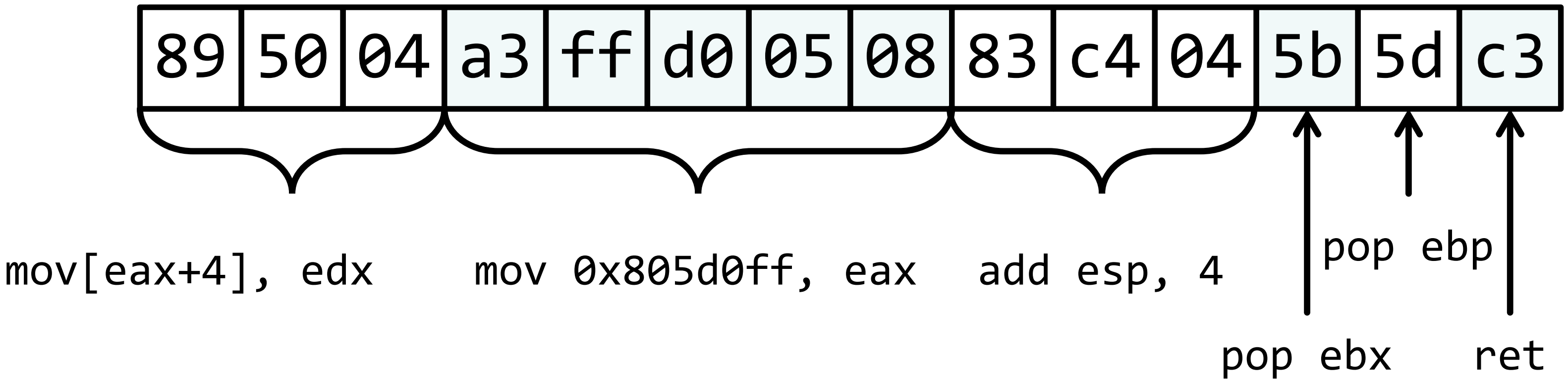
```
pop ecx
ret
```

```
pop ebx
ret
```

Gadget Extraction

89	50	04	a3	ff	d0	05	08	83	c4	04	5b	5d	c3

Gadget Extraction



89 50 04 a3 ff d0 05 08 83 c4 04 5b 5d c3

mov[eax+4], edx

mov 0x805d0ff, eax

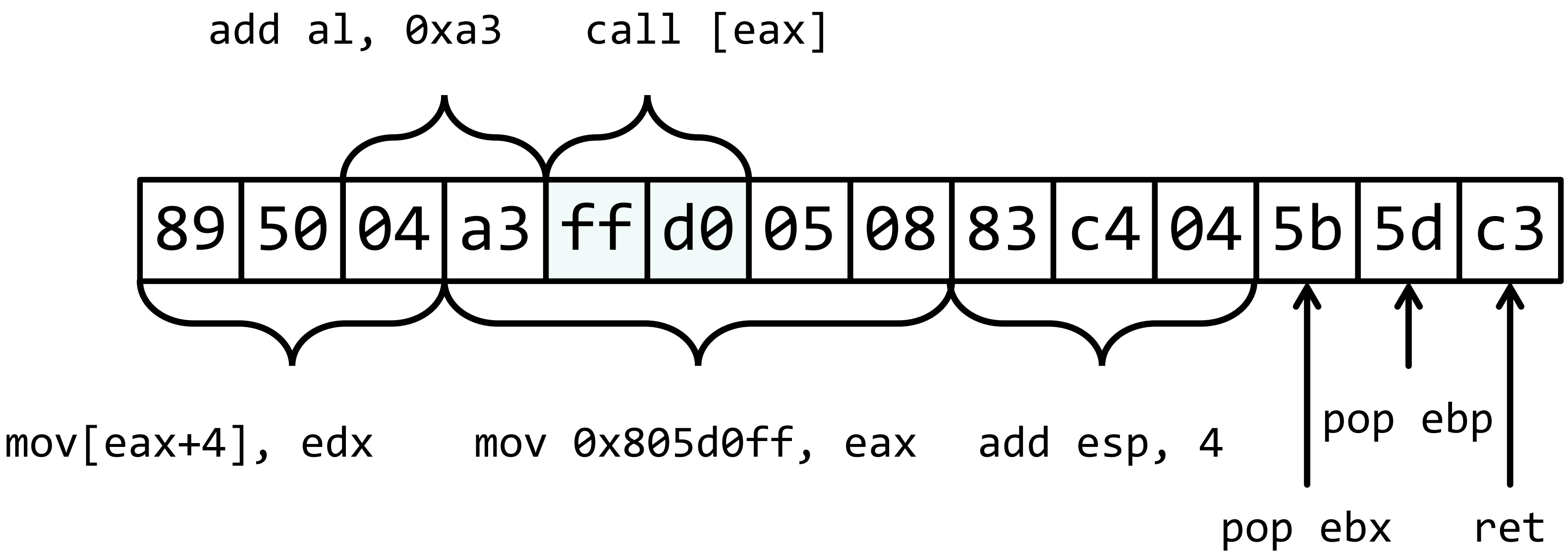
add esp, 4

pop ebp

pop ebx

ret

Gadget Extraction



add al, 0xa3

call [eax]

89	50	04	a3	ff	d0	05	08	83	c4	04	5b	5d	c3

mov[eax+4], edx

mov 0x805d0ff, eax

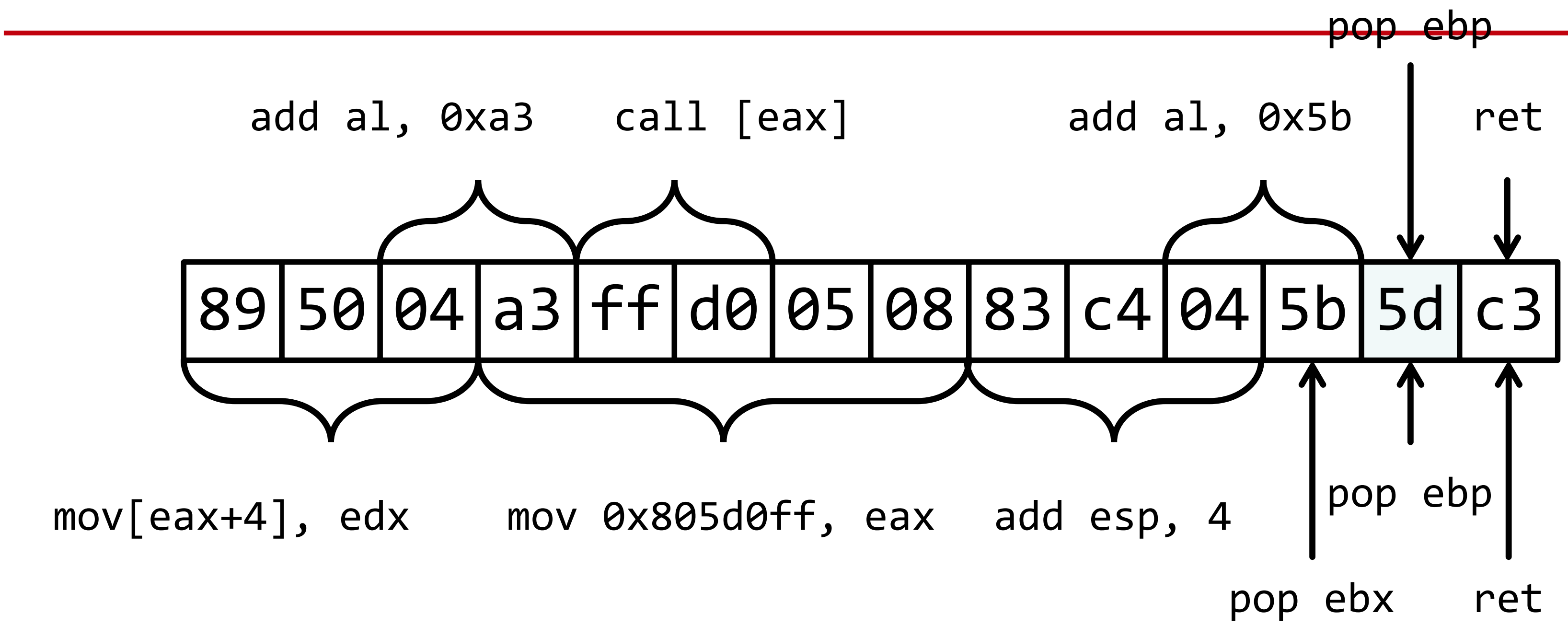
add esp, 4

pop ebp

pop ebx

ret

Gadget Extraction



ROP

- Works against virtually every architecture
- Useful in many situations
  - Non-executable memory regions
  - Signed code
- When combined with memory disclosure vulnerabilities, ROP is very difficult to defend against

ROP Defenses

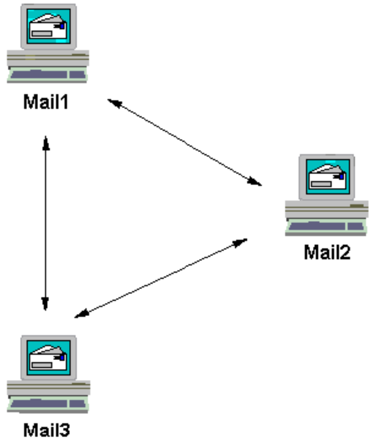
1. Enforcing control flow, stack FIFO characteristics
   – Stackghost, ROPDefender, program shepherding, CFI
2. Detecting abnormal ret frequency
   – DROP, DynIMA
3. Deterministic gadget removal during compilation
   – Gfree
4. Randomized binaries
   – Binary stirring

Internet Services Security

SMTP

Simple Mail Transfer Protocol (SMTP)

- initially specified in RFC 821
- de facto standard for email transmission
- simple, text-based protocol
- MIME used to encode binary files (attachments)
- listens on port 25
- push protocol (used to exchange emails between servers)
- clients have to retrieve emails via other protocols such as IMAP or POP



SMTP Session

```
S: 220 www.example.com ESMTP Postfix
C: HELO mydomain.com
S: 250 Hello mydomain.com
C: MAIL FROM: sender@mydomain.com
S: 250 Ok
C: RCPT TO: friend@example.com
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Subject: test message
C: From: sender@mydomain.com
C: To: friend@example.com
C:
C: Hello,
C: This is a test.
C: Goodbye.
C: .
S: 250 Ok: queued as 12345
C: QUIT
S: 221 Bye
```

SMTP

- Security Issues
  - mail servers have wide distribution base and are publicly accessible
    - software vulnerabilities
    - configuration errors
  - `sendmail`
    - one of the first SMTP implementations (MTAs)
    - long history of vulnerabilities
    - complicated configuration (M4 macro language)
    - e.g., buffer overflow in Sendmail 8.12.9 and before (2003)
  - `postfix, qmail`
    - secure replacements
  - no authentication of sender is performed
    - huge problem
    - makes unsolicited email such a problem

SMTP

- Lack of authentication
  - everyone can connect to a SMTP server and transmit a message
  - server cannot check sender identity (besides IP address)

- Mail relay
  - server accepts message that does not *appear* to be either for a local address or from a local sender

- Solutions for authentication
  - SMTH-AUTH
    - access control list with explicit login
    - clients must be aware of SMTP-AUTH
  - POP-before-SMTP
    - logins are simulated by POP request (which require a login)
    - when a client performs a POP request, its IP address is authenticated with the SMTP server for some time (e.g., 30 minutes)

Spam

- Unsolicited email message

- Gather destination email addresses
  - brute force guessing
  - harvesting (web pages, mailing lists, news groups, …)
  - verified address are more valuable (social engineering, web bug)

- Delivering spam messages
  - own machine (not very smart)
  - other machines
    - open mail relays
    - open proxies
    - web forms
    - zombie nets (compromised machines)

Spam

- Countermeasures
  - client
    - filter tools (e.g., SpamAssassin)
    - automatic report systems
  - blacklists
    - identify origins of spam messages and quickly distribute this information
  - infrastructure
    - SPF (sender policy framework)
    - works by adding “reverse MX” records for a domain
    - only listed machines can send email from this domain

Spam

- Reasons for spam
  - legitimate businesses advertise products and services
  - attempts to get money from victims
    - actually quite old idea, was done with letters decades ago
    - victims sometimes even travel to remote places
  - offer of pornography or other interesting material to lure people on sites where Trojan horses can be installed

- Statistics
  - Ikarus Scan Centers
    - 10 million mail messages per day
    - 60% of these messages are spam
    - 30% contain virus attachments
  - MessageLabs (used by EU)
    - 66% are spam