マイクロアーキテクチャ攻撃演習 1

九州大学　サイバーセキュリティセンター
谷本 輝夫

演習の概要

- Spectre を実際にプロセッサシミュレータで実行
- プロセッサ内で命令が実行される様子を実際に見て、攻撃の仕組みをより深く理解する

今回の内容

- 今回は準備編
  - プロセッサシミュレータを実行できるように準備を行う
- 主な内容
  1. Linux 仮想マシンの起動
  2. Docker のインストール
  3. ビルド済みシミュレータのdocker imageの入手、起動
  4. シミュレータの動作確認
- 時間に余裕がないので困ったらすぐに聞いてください

実験環境

- Kyushu University Educational ENvironment Services (QUEENS)
  - https://queens.s.kyushu-u.ac.jp/user/users/login
  - 要SSO-KID
  - Linux 環境を作成して利用
- プロセッサシミュレータ：gem5
  - http://gem5.org/Main_Page
  - C++で記述されたサイクル精度のプロセッサシミュレータ
  - オープンソース、オープン開発
  - リソース的にQUEENSではビルドできないのでビルド済みdockerイメージを利用
- Dockerって？
  - https://www.docker.com/
  - コンテナと呼ばれるOSレベルの仮想化環境を作成するツール
  - ホストOSとカーネルを共有する仮想環境

1. Linux仮想マシンの起動

- 手順に沿って実施
  - https://queens.s.kyushu-u.ac.jp/user/manuals/linux
  - Linux作成、Linux接続 まで実施してください
  - 秘密鍵のダウンロードリンクは画面下のほうにあります
  - MacOS の人は秘密鍵のパーミッションを変更する必要があります
    - `$ chmod 600 </path/to/private_key>`
  - TeraTerm の使い方が分からない人は以下を参照
    http://ttssh2.osdn.jp/manual/ja/usage/ssh.html

2. Docker のインストール

- 以下のコマンドを実行（sudoから入力）
- Dockerインストール
  - `$ sudo yum install -y docker`
  - 確認：`$ rpm -qa | grep docker`

```
$ rpm -qa | grep docker
docker-18.06.1ce-10.32.amzn1.x86_64
```

- Dockerサービス起動
  - `$ sudo service docker start`
  - 確認：`$ sudo service docker status`

```
$ sudo service docker status
docker (pid  3076) is running...
```

3. ビルド済みシミュレータのdocker imageの入手、起動

- Dockerイメージのダウンロード
  - `$ sudo docker pull teruo41/gem5-spectre:latest`
  - しばらく時間がかかります
  - 確認：`$ sudo docker images`

```
$ sudo docker images
REPOSITORY            TAG                 IMAGE ID            CREATED             SIZE
teruo41/gem5-spectre  latest              6ce20e5cc32a        43 hours ago        1.84GB
```

Dockerを使った作業 1

- Dockerイメージの起動
  - `$ sudo docker run -i -t teruo41/gem5-spectre:latest`
    - `-i, --interactive`
    - `-t, --tty`
  - 様子：

```
$ sudo docker run -i -t teruo41/gem5-spectre:latest
[gem5user@deaaf0fac9de /]$
```

- Docker内での作業終了
  - `$ exit`
  - 確認：`$ sudo docker ps -a`

```
$ sudo docker ps -a
CONTAINER ID        IMAGE                          COMMAND             CREATED             STATUS                      PORTS               NAMES
deaaf0fac9de        teruo41/gem5-spectre:latest    "/bin/bash"         12 minutes ago      Exited (0) 10 minutes ago                       serene_stonebraker
```

Dockerを使った作業 2

- Docker内での作業再開
  - `$ sudo docker start -i <コンテナ名>`
  - コンテナ名は `$ sudo docker ps -a` で確認

4. シミュレータの動作確認

- Dockerコンテナ内で以下を実行
  - `$ cd /home/gem5user/gem5-spectre`
  - `$ gem5/build/X86/gem5.opt -d gem5out/runtest2 gem5/configs/learning_gem5/part1/two_level_o3ltage.py`

- 結果

```
$ gem5/build/X86/gem5.opt -d gem5out/runtest2
gem5/configs/learning_gem5/part1/two_level_o3ltage.py
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Jul 14 2019 18:24:43
gem5 started Aug  6 2019 08:22:43
gem5 executing on deaaf0fac9de, pid 20
command line: gem5/build/X86/gem5.opt -d gem5out/runtest2
gem5/configs/learning_gem5/part1/two_level_o3ltage.py

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range
assigned (512 Mbytes)
0: system.remote_gdb: listening for remote gdb on port 7000
Beginning simulation!
info: Entering event queue @ 0.  Starting simulation...
Hello world!
Exiting @ tick 31452000 because exiting with last active thread context
```

- できるはずのファイル
  （/home/gem5user/gem5-spectre/gem5out/runtest2/）
  - config.ini　　動かしたシミュレータの設定内容
  - config.json　　動かしたシミュレータの設定内容
  - stats.txt　　シミュレータの動作結果の統計情報

- 次回この環境を使ってSpectreの動作を解析します

- 早く終わった人はプログラムを作って実行してみてください
  - gem5 実行コマンドの最後に実行ファイルを指定すると任意のプログラムを実行できます（この場合引数は与えられません）

Dockerイメージについて

- Dockerfile（イメージの設計書）をGithubで公開
  - https://github.com/teruo41/gem5-spectre/blob/master/Dockerfile
  - 得体のしれないイメージを動かすのが不安な人は確認してください
  - （ちなみに、DockerHubでビルドすると３時間ほどかかります）

- おつかれさまでした