

マイクロアーキテクチャ攻撃演習 1

九州大学 大学院システム情報科学研究所

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. Docker のインストール
 2. ビルド済みシミュレータのdocker imageの入手
 3. Docker を使った作業 1
 4. Docker を使った作業 2
 5. シミュレータの動作確認

実験環境

- ▶ 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
 - ▶ ビルド済み環境が入った Docker image を公開
<https://hub.docker.com/repository/docker/teruo41/gem5-spectre>
- ▶ 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. Docker 環境作成

- ▶ DockerのWebページを参考にインストールしてください
<https://docs.docker.com/get-docker/>
 - ▶ Windows, MacOS の場合は、Docker Desktop の利用が便利
<https://docs.docker.com/desktop/>
 - ▶ System Requirements をしっかり確認してください
- ▶ Windows の HOME 版などでは WSL 2 を使って動かすことも可能
 - ▶ WSL (Windows Subsystem Linux)
<https://docs.microsoft.com/ja-jp/windows/wsl/>
 - ▶ Hyper-v
<https://docs.microsoft.com/ja-jp/virtualization/hyper-v-on-windows/>

2. ビルド済みシミュレータの docker imageの入手 (Linux)

▶ Dockerイメージのダウンロード

▶ コマンド (Linux) :

```
$ sudo docker pull teruo41/gem5-spectre:latest
```

▶ しばらく時間がかかります

▶ 確認 (Linux) :

```
$ sudo docker images
```

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

3. 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 (Linux)

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

▶ Docker内での作業終了

▶ `$ exit`

▶ 確認 (Linux) : `$ 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
```

4. 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

5. 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
- ▶ 次はこの環境を使ってSpectreの動作を解析します
- ▶ 好きなプログラムを作って実行してみてください
 - ▶ 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）
 - ▶ この場合、シングルスレッドプログラムのみ実行可能です

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）