

TCSS 562:
SOFTWARE ENGINEERING
FOR CLOUD COMPUTING

AWS Overview/Demo
Cloud Enabling Technology

Wes J. Lloyd
School of Engineering and Technology
University of Washington – Tacoma



1

OFFICE HOURS – FALL 2025

■ **Thursdays:**

■ 6:00 to 7:00 pm - CP 229 & Zoom

■ **Friday – *** THIS WEEK *****

■ **12:30 pm to 1:30 pm – ONLINE via Zoom**

■ Or email for appointment

➢ Office Hours set based on Student Demographics survey feedback

➢ * - Friday office hours may be adjusted or canceled due meeting conflicts or other obligations. Adjustments will be announced via Canvas.

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)
School of Engineering and Technology, University of Washington – Tacoma

L12.2

2

OBJECTIVES – 11/6

■ **Questions from 11/4**

■ Tutorials Questions

■ Class Presentations:
Cloud Technology or Research Paper Review

■ Ch. 5: Cloud Enabling Technology

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)
School of Engineering and Technology, University of Washington – Tacoma

L12.3

3

ONLINE DAILY FEEDBACK SURVEY

■ Daily Feedback Quiz in Canvas – Take After Each Class

■ Extra Credit for completing

Announcements

Assignments

Discussions

Zoom

Grades

People

Pages

Files

Quizzes

Collaborations

UW Libraries

UW Resources

Upcoming Assignments

Class Activity 1 – Implicit vs. Explicit Parallelism
Available until Oct 13 at 11:59pm | Due Oct 7 at 7:59pm | ~15 pts

Tutorial 1 - Linux
Available until Oct 19 at 11:59pm | Due Oct 13 at 11:59pm | ~20 pts

Past Assignments

TCSS 562 - Online Daily Feedback Survey - 10/5
Available until Oct 18 at 11:59pm | Due Oct 6 at 8:59pm | ~15 pts

TCSS 562 - Online Daily Feedback Survey - 9/30
Available until Oct 18 at 11:59pm | Due Oct 4 at 8:59pm | ~15 pts

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)
School of Engineering and Technology, University of Washington – Tacoma

L12.4

4

TCSS 562 - Online Daily Feedback Survey - 10/5

Started: Oct 7 at 1:13am

Quiz Instructions

Question 1

0.5 pts

On a scale of 1 to 10, please classify your perspective on material covered in today's class:

1

2

3

4

5

6

7

8

9

10

Mostly Review To Me

Equal New and Review

Mostly New To Me

Question 2

0.5 pts

Please rate the pace of today's class:

1

2

3

4

5

6

7

8

9

10

Slow

Just Right

Fast

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)
School of Engineering and Technology, University of Washington – Tacoma

L12.5

5

MATERIAL / PACE

■ Please classify your perspective on material covered in today's class (43 respondents, 28 in-person, 15 online):

■ 1-mostly review, 5-equal new/review, 10-mostly new

■ **Average – 6.36 (↓ - previous 6.55)**

■ Please rate the pace of today's class:

■ 1-slow, 5-just right, 10-fast

■ **Average – 5.19 (↑ - previous 5.18)**

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)
School of Engineering and Technology, University of Washington – Tacoma

L12.6

6

Slides by Wes J. Lloyd

L12.1

FEEDBACK FROM 11/4

- **Unclear point:**
You should put a backslash behind double-quotation marks when assigning variables to JSON in BASH scripts
 - To load JSON objects into BASH variables "escaping" double-quote marks is required
 - This is because **JSON requires double quotes** for both keys and values. JSON does not support single quotes.
- ```
json={ "name": "Susan\u0020Smith" }
```
- Highlighted chars put the " in the variable
  - Unlimited quotes define keys and values

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.7

7

TERM PROJECT PROPOSALS

- 14 Total term project proposals received
- Project proposals are under review
  - 6 remaining proposals to review
- If asked for minor revisions, can simply address item(s) and submit a brief update
- Status & Feedback to be provided:
  - proposals accepted, or
  - revisions requested

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.8

8

OBJECTIVES – 11/6

- Questions from 11/4
- **Tutorials Questions**
- Class Presentations:  
Cloud Technology or Research Paper Review
- Ch. 5: Cloud Enabling Technology

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.9

9

TUTORIAL 4 – DUE NOV 11

- Introduction to AWS Lambda with the Serverless Application Analytics Framework (SAAF)  
[https://faculty.washington.edu/wlloyd/courses/tcss562/tutorials/TCSS462\\_562\\_f2025\\_tutorial\\_4.pdf](https://faculty.washington.edu/wlloyd/courses/tcss562/tutorials/TCSS462_562_f2025_tutorial_4.pdf)
- Set up Java development environment
- Introduction to Maven build files for Java
- Create and Deploy "hello" Java AWS Lambda Function
- Create API Gateway REST endpoint
- Sequential testing of "hello" AWS Lambda Function
  - API Gateway endpoint, AWS Lambda CLI Function invocation, AWS Function URL
- Profiling function performance with SAAF
- Concurrent function testing with faas\_runner
- Performance analysis using faas\_runner reports
- Two function pipeline development task: Caesar Cipher

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.10

10

TUTORIAL 5 – DUE NOV 14

- Introduction to Lambda II: Working with Files in S3, Cloud Trail, and Amazon Event Bridge Rules
- [https://faculty.washington.edu/wlloyd/courses/tcss562/tutorials/TCSS462\\_562\\_f2025\\_tutorial\\_5.pdf](https://faculty.washington.edu/wlloyd/courses/tcss562/tutorials/TCSS462_562_f2025_tutorial_5.pdf)
- Customize the Request object (add getters/setters)
  - Why do this instead of HashMap ?
- Import dependencies (jar files) into project for AWS S3
- Create an S3 Bucket
- Give your Lambda function(s) permission to work with S3
- Write to the CloudWatch logs
- Use of CloudTrail to generate S3 events
- Creating Event Bridge rule to capture events from CloudTrail
- Have the Event Bridge rule trigger a Lambda function with a static JSON input object (hard-coded filename)
- **Optional:** for the S3 PutObject event, dynamically extract the name of the file put to the S3 bucket for processing

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.11

11

TUTORIAL 6

- Introduction to Lambda III: Serverless Databases
- To be posted...
- Create and use SQLite databases using sqlite3 tool
- Deploy Lambda function with SQLite3 database under /tmp
- Compare in-memory vs. file-based SQLite DBs on Lambda
- Create an Amazon Aurora "Serverless" v2 MySQL database
- Using an ec2 instance in the same VPC (Region + availability zone) connect and interact with the database using the mysql CLI app
- Deploy an AWS Lambda function that uses the MySQL "serverless" database

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.12

12

## L12.13

L12.14

L12.16

L12.1

18

AWS EC2

- Elastic Compute Cloud
- Instance types: <https://instances.vantage.sh/>
  - On demand Instance – full price
  - Reserved Instance – contract based where customer guarantees VM rental for a fixed period of time (e.g. 1 year, 3 years, etc.)  
Deeper discounts with longer term commitments
  - Spot Instance – portion of cloud capacity reserved for low cost instances, when demand exceeds supply instances are randomly terminated with 2-minute warning
    - Users can make diverse VM requests using different types, zones, regions, etc. to minimize instance terminations
    - Developers can design for failure because often only 1 or 2 VMs in a cluster fail at any given time. They then need to be replaced.
  - Dedicated host – reserved private HW (server)
  - Instance families –
    - General, compute-optimized, memory-optimized, GPU, etc.

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.19

19

AWS EC2 - 2

- Storage types
  - Instance storage - ephemeral storage
    - Temporary disk volumes stored on disks local to the VM
    - Evolution: physical hard disk drives (HDDs)
    - Solid state drives (SSDs)
    - Non-volatile memory express (NVMe) drives (closer to DRAM speed)
  - EBS - Elastic block store
    - Remotely hosted disk volumes
  - EFS - Elastic file system
    - Shared file system based on network file system
    - VMs, Lambdas, Containers mount/interact with shared file system
    - Somewhat expensive

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.20

20

INSTANCE STORAGE

- Also called ephemeral storage
- Persisted using images saved to S3 (simple storage service)
  - ~2.3¢ per GB/month on S3
  - 5GB of free tier storage space on S3
- Requires “burning” an image
- Multi-step process:
  - Create image files
  - Upload chunks to S3
  - Register image
- Launching a VM
  - Requires downloading image components from S3, reassembling them... is potentially slow
- VMs with instance store backed root volumes not pause-able
- Historically root volume limited to 10-GB max- **faster imaging...**

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.21

21

ELASTIC BLOCK STORE

- EBS provides 1 drive to 1 virtual machine (**1 : 1 (not shared)**)
- EBS cost model is different than instance storage (uses S3)
  - ~10¢ per GB/month for General Purpose Storage (GP2)
  - ~8¢ per GB/month for General Purpose Storage (GP3)
  - 30GB of free tier storage space
- EBS provides “live” mountable volumes
  - Listed under volumes
  - Data volumes:** can be mounted/unmounted to any VM, dynamically at any time
  - Root volumes:** hosts OS files and acts as a boot device for VM
  - In Linux drives are linked to a mount point “directory”
- Snapshots back up EBS volume data to S3
  - Enables replication (required for horizontal scaling)
  - EBS volumes not actively used should be snapshotted, and deleted to save EBS costs...

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.22

22

EBS VOLUME TYPES - 2

- Metric: I/O Operations per Second (IOPS)
- General Purpose 2 (GP2)
  - 3 IOPS per GB, min 100 IOPS (<34GB), max of 16,000 IOPS
  - 250MB/sec throughput per volume
- General Purpose 3 (GP3 – new Dec 2020)
  - Max 16,000 IOPS, Default 3,000 IOPS
  - GP2 requires creating a 1TB volume to obtain 3,000 IOPS
  - GP3 all volumes start at 3000 IOPS and 125 MB/s throughput
  - 1000 additional IOPS beyond 3000 is \$5/month up to 16000 IOPS
  - 125 MB/s additional throughput is \$5/month up to 1000 MB/s throughput

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.23

23

EBS VOLUME TYPES - 3

- Provisioned IOPS (IO1)
  - Legacy, associated with GP2
  - Allows user to create custom disk volumes where they pay for a specified IOPS and throughput
  - 32,000 IOPS, and 500 MB/sec throughput per volume MAX
- Throughput Optimized HDD (ST1)
  - Up to 500 MB/sec throughput
  - 4.5 ¢ per GB/month
- Cold HDD (SC1)
  - Up to 250 MB/sec throughput
  - 2.5 ¢ per GB/month
- Magnetic
  - Up to 90 MB/sec throughput per volume
  - 5 ¢ per GB/month

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.24

24

ELASTIC FILE SYSTEM (EFS)

- EFS provides 1 volume to many client (1 : n) **shared storage**
- Network file system (based on NFSv4 protocol)
- Shared file system for EC2, Fargate/ ECS, Lambda
- Enables mounting (sharing) the same disk "volume" for R/W access across multiple instances at the same time
- Different performance and limitations vs. EBS/Instance store
- Implementation uses abstracted EC2 instances
- ~ 30 ¢ per GB/month storage – **default burstable throughput**
- Throughput modes:**
  - Can modify modes only once every 24 hours
- Burstable Throughput Model:**
  - Baseline – 50kb/sec per GB
  - Burst – 100MB/sec per GB (for volumes sized 10GB to 1024 GB)
  - Credits – .72 minutes/day per GB

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.25

25

ELASTIC FILE SYSTEM (EFS) - 2

- Burstable Throughput Rates**
  - Throughput rates: baseline vs burst
  - Credit model for bursting: maximum burst per day

Information subject to revision

| File System Size (GiB) | Baseline Aggregate Throughput (MiB/s) | Burst Aggregate Throughput (MiB/s) | Maximum Burst Duration (Min/Day) | % of Time File System Can Burst (Per Day) |
|------------------------|---------------------------------------|------------------------------------|----------------------------------|-------------------------------------------|
| 10                     | 0.5                                   | 100                                | 7.2                              | 0.5%                                      |
| 256                    | 12.5                                  | 100                                | 180                              | 12.5%                                     |
| 512                    | 25.0                                  | 100                                | 360                              | 25.0%                                     |
| 1024                   | 50.0                                  | 100                                | 720                              | 50.0%                                     |
| 1536                   | 75.0                                  | 150                                | 720                              | 50.0%                                     |
| 2048                   | 100.0                                 | 200                                | 720                              | 50.0%                                     |
| 3072                   | 150.0                                 | 300                                | 720                              | 50.0%                                     |
| 4096                   | 200.0                                 | 400                                | 720                              | 50.0%                                     |

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.26

26

ELASTIC FILE SYSTEM (EFS) - 3

- Throughput Models**
  - Provisioned Throughput Model**
- For applications with: high performance requirements, but low storage requirements
- Get high levels of performance w/o overprovisioning capacity
- \$6 MB/s-Month (Virginia Region)
  - Default is 50kb/sec for 1 GB, .05 MB/s = 30 ¢ per GB/month
- If file system metered size has higher baseline rate based on size, file system follows default Amazon EFS Bursting Throughput model
  - No charges for Provisioned Throughput below file system's entitlement in Bursting Throughput mode
  - Throughput entitlement = 50kb/sec per GB

Information subject to revision

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.27

27

ELASTIC FILE SYSTEM (EFS) - 4

Performance Comparison, Amazon EFS and Amazon EBS

|                       | Amazon EFS               | Amazon EBS Provisioned IOPS |
|-----------------------|--------------------------|-----------------------------|
| Per-operation latency | Low, consistent latency. | Lowest, consistent latency. |
| Throughput scale      | 10+ GB per second.       | Up to 2 GB per second.      |

Storage Characteristics Comparison, Amazon EFS and Amazon EBS

|                             | Amazon EFS                                                                                                 | Amazon EBS Provisioned IOPS                                                 |
|-----------------------------|------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| Availability and durability | Data is stored redundantly across multiple AZs.                                                            | Data is stored redundantly in a single AZ.                                  |
| Access                      | Up to thousands of Amazon EC2 instances, from multiple AZs, can connect concurrently to a file system.     | A single Amazon EC2 instance in a single AZ can connect to a file system.   |
| Use cases                   | Big data and analytics, media processing workflows, content management, web serving, and home directories. | Boot volumes, transactional and NoSQL databases, data warehousing, and ETL. |

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.28

28

EC2

- EC2 Spot Instance Advisor:**
- <https://aws.amazon.com/ec2/spot/instance-advisor/>
- Provides sortable list of ec2 instance types with interruption (termination) frequencies
- Helps you choose an instance type that is less likely to be terminated
- Best practices for using spot instances:
  - <https://docs.aws.amazon.com/whitepapers/latest/cost-optimization-leveraging-ec2-spot-instances/spot-best-practices.html>

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.29

29

EC2 - 2

- On Amazon EC2, what is a "metal" Instance?**
- A bare metal server is not shared with anyone
- There is no virtualization hypervisor (program the contextualizes and hosts virtual machines)
- The operating system is installed directly on the root disk and the machine is booted directly like a laptop or desktop computer
- The user can install any operating system and make configurations changes to the machine's base operating system
- The user can then install and control a virtualization hypervisor on bare metal servers
- Bare metal servers were offered on AWS starting in ~2017

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.30

30

AMAZON MACHINE IMAGES

- AMIs
- Unique for the operating system (root device image)
- Two types
  - Instance store
  - Elastic block store (EBS)
- Deleting requires multiple steps
  - Deregister AMI
  - Delete associated data - (files in S3)
- Forgetting both steps leads to costly "orphaned" data
  - No way to instantiate a VM from deregistered AMIs
  - Data still in S3 resulting in charges

31

EC2 VIRTUALIZATION HYPERVISORS

- 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup>, 4<sup>th</sup> generation → XEN-based
- 5<sup>th</sup>+ generation Instances → AWS Nitro virtualization
- XEN (pre-nitro) - two virtualization modes:
  - XEN Para-virtualization (PV) "paravirtual"
    - 10GB Amazon Machine Image - base image size limit
    - Addressed poor performance of old XEN HVM mode
    - I/O performed using special XEN kernel with XEN paravirtual mode optimizations for better performance
    - Requires OS to have an available paravirtual kernel
    - PV VMs: use common **AKI** files on AWS - **Amazon kernel Image(s)**

32

EC2 VIRTUALIZATION - HVM

- XEN HVM (hardware virtualization) mode:
  - HVM just meant that virtualization used CPU extensions (Intel VT-x or AMD-V) to run an unmodified OS directly on the VM
    - Today HVM usually implies there is more than just native-CPU virtualization support, but also I/O-device support, etc.
  - Full virtualization - no special OS kernel required (no AKI)
  - Computer entirely simulated (... i.e. slow)
  - MS Windows pre-nitro ran in "HVM" mode
  - Allows work around: 10GB instance store root volume limit
  - Kernel is on the root volume (under /boot)
  - No AKIs (kernel images)
  - Commonly used today (EBS-backed instances)

33

XEN PV VS HVM ON EC2

| Feature                     | Paravirtual (PV)                                                                                                                 | Hardware Virtual Machine (HVM)                                                                                                                                     |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Guest OS Modification       | Requires the guest OS to be modified (aware of the hypervisor).                                                                  | Allows the use of unmodified guest OS, as if running on bare metal hardware.                                                                                       |
| Boot Method                 | Boots with a special boot loader called PV-GRUB, which then chain loads the kernel.                                              | Boots by executing the master boot record of the root block device, like a physical machine.                                                                       |
| Hardware Access             | Relies on a patched kernel and specialized drivers (hypercalls) for I/O operations and direct communication with the hypervisor. | Emulates a full set of hardware and takes advantage of hardware virtualization extensions (Intel VT-x, AMD-V) for fast access to the underlying physical hardware. |
| Performance (Historically)  | Historically offered better performance than early HVM instances due to the use of direct drivers for network/storage I/O.       | Early HVM had more overhead due to hardware emulation.                                                                                                             |
| Performance (Modern Legacy) | Performance is limited and lacks modern enhancements.                                                                            | With the availability of PV drivers for HVM AMIs and hardware advancements, HVM instances provide better or superior performance for most workloads.               |
| Feature Support             | Does not support modern features like enhanced networking or GPU processing.                                                     | Required for advanced features such as enhanced networking, GPU instances, and specific instance types (e.g., i2 instances).                                       |

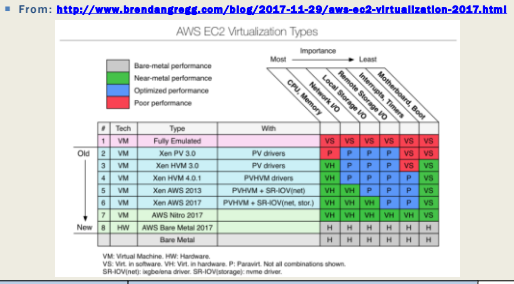
34

EC2 VIRTUALIZATION - NITRO

- Nitro based on Kernel-based-virtual-machines
  - Stripped down version of Linux KVM hypervisor
  - Uses KVM core kernel module
  - I/O access has a direct path to the device
- Goal: provide indistinguishable performance from bare metal
- There are 5 Nitro versions (v2 to v6)
- Article describes features and provides links to instance families (m, c, r, etc.) where it is possible to check the Nitro version for specific instance types to understand feature evolution
- <https://docs.aws.amazon.com/ec2/latest/instancetypes/ec2-nitro-instances.html>

35

EVOLUTION OF AWS VIRTUALIZATION



36

### INSTANCE ACTIONS

- Stop
  - Costs of "pausing" an instance
- Terminate
- Reboot

- Image management
- Creating an image
  - EBS (snapshot)
- Bundle image
  - Instance-store

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.37

37

### EC2 INSTANCE: NETWORK ACCESS

- Public IP address
- Elastic IPs
  - Costs: in-use FREE, not in-use ~12 \$/day
  - Not in-use (e.g. "paused" EBS-backed instances)
- Security groups
  - E.g. firewall
- Identity access management (IAM)
  - AWS accounts, groups
- VPC / Subnet / Internet Gateway / Router
- NAT-Gateway

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.38

38

### SIMPLE VPC

- Recommended when using Amazon EC2

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.39

39

### VPC SPANNING AVAILABILITY ZONES

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.40

40

### INSPECTING INSTANCE INFORMATION

- EC2 VMs run a local metadata service
- Can query instance metadata to self discover cloud config attributes
- Version 2 (default) of the metadata service requires a token**
- Get Token:
 

```
TOKEN=$(curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600")
```
- Find your instance ID:
 

```
curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/instance-id
```
- See: <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/configuring-instance-metadata-service.html#instance-metadata-retrieval-examples>

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.41

41

### SIMPLE STORAGE SERVICE (S3)

- Key-value blob storage
- What is the difference vs. key-value stores (NoSQL DB)?
- Can mount an S3 bucket as a volume in Linux
  - Supports common file-system operations
- Provides eventual consistency
- Can store Lambda function state for life of container.

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.42

42

AWS CLI

- Launch Ubuntu VM
  - Instances | Launch Instance
- Install the general AWS CLI
  - sudo apt install awscli
- Create config file [default]

```
aws_access_key_id = <access key id>
aws_secret_access_key = <secret access key>
region = us-east-1
```

November 6, 2025

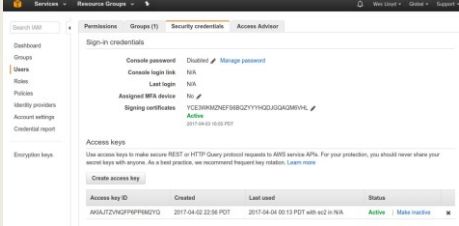
TCSS462/562 (Software Engineering for) Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.43

43

AWS CLI - 2

- Creating access keys: IAM | Users | Security Credentials | Access Keys | Create Access Keys



November 6, 2025

TCSS462/562 (Software Engineering for) Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.44

44

AWS CLI - 3

- Export the config file
  - Add to /home/ubuntu/.bashrc

```
export AWS_CONFIG_FILE=$HOME/.aws/config
```
- Try some commands:
  - aws help
  - aws command help
  - aws ec2 help
  - aws ec2 describe-instances --output text
  - aws ec2 describe-instances --output json
  - aws s3 ls
  - aws s3 ls vmscaleruw

November 6, 2025

TCSS462/562 (Software Engineering for) Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.45

45

LEGACY / SERVICE SPECIFIC CLI(S)

- sudo apt install ec2-api-tools
- Provides more concise output
- Additional functionality
- Define variables in .bashrc or another sourced script:

```
export AWS_ACCESS_KEY={your access key}
export AWS_SECRET_KEY={your secret key}
```
- ec2-describe-instances
- ec2-run-instances
- ec2-request-spot-instances
- EC2 management from Java:  
<http://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/index.html>
- Some AWS services have separate CLI installable by package

November 6, 2025

TCSS462/562 (Software Engineering for) Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.46

46

AMI TOOLS

- Amazon Machine Images tools
- For working with disk volumes
- Can create live copies of any disk volume
  - Your local laptop, ec2 root volume (EBS), ec2 ephemeral disk
- Installation:  
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ami-tools-commands.html>
- AMI tools reference:  
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ami-tools-commands.html>
- Some functions may require private key & certificate files

November 6, 2025

TCSS462/562 (Software Engineering for) Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.47

47

PRIVATE KEY AND CERTIFICATE FILE

- Install openssl package on VM

```
generate private key file
$openssl genrsa 2048 > mykey.pk

generate signing certificate file
$openssl req -new -x509 -nodes -sha256 -days 36500 -key mykey.pk -outform PEM -out signing.cert
```
- Add signing.cert to IAM | Users | Security Credentials | - - new signing certificate - -
- From: [http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/setup-ami-tools.html?icpid=docs\\_iam\\_console#ami-tools-create-certificate](http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/setup-ami-tools.html?icpid=docs_iam_console#ami-tools-create-certificate)

November 6, 2025

TCSS462/562 (Software Engineering for) Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.48

48

PRIVATE KEY, CERTIFICATE FILE

- These files, combined with your AWS\_ACCESS\_KEY and AWS\_SECRET\_KEY and AWS\_ACCOUNT\_ID enable you to publish new images from the CLI
- Objective:
  - Configure VM with software stack
  - Burn new image for VM replication (**horizontal scaling**)
- An alternative to bundling volumes and storing in S3 is to use a containerization tool such as Docker. . .
- Create image script . . .

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.49

49

SCRIPT: CREATE A NEW INSTANCE STORE  
IMAGE FROM LIVE DISK VOLUME

```
image=$1
echo "burn image $image"
echo "$image" > image.id
mkdfr /mnt/tmp
AWS_KEY_DIR=/home/ubuntu/.aws
export EC2_URL=http://ec2.amazonaws.com
export S3_URL=https://s3.amazonaws.com
export EC2_PRIVATE_KEY=${AWS_KEY_DIR}/mykey.pk
export EC2_CERT=${AWS_KEY_DIR}/signing.cert
export AWS_USER_ID={your account id}
export AWS_ACCESS_KEY={your aws access key}
export AWS_SECRET_KEY={your aws secret key}
ec2-bundle-vol -s 5000 -u ${AWS_USER_ID} -c ${EC2_CERT} -k ${EC2_PRIVATE_KEY}
--ec2cert /etc/ec2/amiutils/cert-ec2.pem --no-inherit -r x86_64 -p $image -i
/etc/ec2/amiutils/cert-ec2.pem
cd /tmp
ec2-upload-bundle -b tcss562 -m $image.manifest.xml -a ${AWS_ACCESS_KEY} -s
${AWS_SECRET_KEY} --url http://s3.amazonaws.com --location US
ec2-register tcss562/$image.manifest.xml --region us-east-1 --kernel aki-
88aa75e1
```

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.50

50

MAKE A DISK FROM AN IMAGE FILE

```
***** ON THE LOCAL COMPUTER *****
create 1200 MB virtual disk = 1,258,291,200 bytes
sudo dd if=/dev/zero of=vhd.img bs=1M count=1200
format the disk using the ext4 filesystem
sudo mkfs.ext4 vhd.img
mount the disk at "/mnt"
sudo mount -t auto -o loop vhd.img /mnt
check that the disk is mounted
df -h
create a hello file (or copy data) to the new virtual disk
cd /mnt
sudo echo "hello world !" > hello.txt
ls -l
cd
unmount the virtual disk
sudo umount /mnt
```

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.51

51

COMPRESS IMAGE, PUSH TO S3

```
compress the disk
bzip2 vhd.img

push the disk image to S3
aws s3 cp vhd.img.bz2 s3://tcss562-f21-images
```

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.52

52

RESTORE ON THE CLOUD

```
***** ON THE AWS EC2 VM *****
with the awscli installed and configured
download the image from S3
aws s3 cp s3://tcss562-f21-images/vhd.img.bz2 vhd.img.bz2
uncompress the image
bzip2 -d vhd.img.bz2
we need to calculate the number of sectors for the
partition
disk sectors are 512 bytes each
divide the disk size by 512 to determine sectors
sectors = 1258291200 / 512 = 2459648
create a disk partition for this disk that is
2459648 sectors in size using the ephemeral drive or
a newly mounted EBS volume that is unformatted
sudo fdisk /dev/nvme1n1
```

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.53

53

PARTITION THE DISK

```
Welcome to fdisk (util-linux 2.34).
Command (m for help): n
Partition type
p primary (0 primary, 0 extended, 4 free)
e extended (container for logical partitions)

Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-97656249, default 2048): 2048
Last sector, +/-sectors or +/-size(K,M,G,T,P) (2048-97656249, default
97656249): 2459648

Created a new partition 1 of type 'Linux' and of size 1.2 GiB.

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 83
Changed type of partition 'Linux' to 'Linux'.

Command (m for help): w (to write and exit)
```

November 6, 2025

TCSS462/562: Software Engineering for Cloud Computing [Fall 2025]  
School of Engineering and Technology, University of Washington - Tacoma

L12.54

54

COPY DATA TO NEW DISK PARTITION

```
now check if the partition has been created.
it should be listed as /dev/nvme1n1p1:
ls /dev/nvme1n1*

now copy the data to the partition
sudo dd if=vhd.img of=/dev/nvme1n1p1

mount the disk
sudo mount /dev/nvme1n1p1 /mnt

and check if the hello file is there
cat /mnt/hello.txt

we were able to copy the disk image to the cloud
and we never had to format the cloud disk
this examples copies a filesystem from a local disk
to the cloud disk
```

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)  
School of Engineering and Technology, University of Washington - Tacoma

L12.55

55

FOR MORE INFORMATION

- Example script:
- <https://faculty.washington.edu/wlloyd/courses/tcss562/examples/copy-disk-to-cloud.sh>
- URLs:
- <https://help.ubuntu.com/community/DriveImaging>
- <https://www.tecmint.com/create-virtual-harddisk-volume-in-linux/>

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)  
School of Engineering and Technology, University of Washington - Tacoma

L12.56

56

COST SAVINGS MEASURES

- **From Tutorial 3:**
- **#1: ALWAYS USE SPOT INSTANCES FOR COURSE/RESEARCH RELATED PROJECTS**
- **#2: NEVER LEAVE AN EBS VOLUME IN YOUR ACCOUNT THAT IS NOT ATTACHED TO A RUNNING VM**
- **#3: BE CAREFUL USING PERSISTENT REQUESTS FOR SPOT INSTANCES**
- **#4: TO SAVE/PERSIST DATA, USE EBS SNAPSHOTS AND THEN**
- **#5: DELETE EBS VOLUMES FOR TERMINATED EC2 INSTANCES.**
- **#6: UNUSED SNAPSHOTS AND UNUSED EBS VOLUMES SHOULD BE PROMPTLY DELETED !!**
- **#7: USE PERSISTENT SPOT REQUESTS AND THE "STOP" FEATURE TO PAUSE VMS DURING SHORT BREAKS**

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)  
School of Engineering and Technology, University of Washington - Tacoma

L12.57

57

OBJECTIVES – 11/6

- Questions from 11/4
- Tutorials Questions
- Class Presentations:
  - Cloud Technology or Research Paper Review
  - **Ch. 5: Cloud Enabling Technology**

November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)  
School of Engineering and Technology, University of Washington - Tacoma

L12.58

58

QUESTIONS



November 6, 2025

TCSS462/562: (Software Engineering for) Cloud Computing (Fall 2025)  
School of Engineering and Technology, University of Washington - Tacoma

L12.11  
6

118