**Identity&Crypto**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 루트 사용자 계정의 액세스 키 제거(1.1.4) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 루트 사용자 계정과 연결된 모든 액세스 키를 제거하는 것을 권장한다. | | |
| 설명 | 루트 사용자 계정과 연결된 액세스 키를 제거할 경우 계정이 손상될 수 있는 공격 벡터를 줄일 수 있다. 루트 액세스 키 제거 및 최소한의 권한을 부여한 계정을 생성하여 사용할 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 루트 계정에 액세스 키가 존재하지 않는 경우   ▶ 취약 : 루트 계정에 액세스 키가 존재하는 경우   ▶ Default Value : False | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/console/ 을 통해 AWS Console에 접속한다. 2. 우측 상단의 계정명을 클릭한다.  3. 드롭다운 메뉴바에서 '내 보안 자격 증명(My Security Credentials)'을 클릭한다.  4. '액세스 키(Access keys)' 탭을 선택한다. 5. 활성화된 액세스 키가 존재할 경우, 비활성화(Make Inactive) 설정 후 삭제를 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '자격 증명 보고서(Credential Report)'를 클릭한다. 3. '보고서 다운로드(Download Report)'를 클릭한다. 4. csv 파일에서 <root\_account> 의 "access\_key\_1\_active" 와 "access\_key\_2\_active" 값이 FALSE로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 "AccountAccessKeysPresent" 항목이 0 으로 설정되어 있는지 확인한다. 루트 액세스 키가 존재하지 않을 경우 0이 출력된다.  # aws iam get-account-summary | grep "AccountAccessKeysPresent" | | |
| Reference | 1.4 Ensure no 'root' user account access key exists (Automated) | | |
| Site | https://docs.aws.amazon.com/general/latest/gr/aws-access-keys-best-practices.html https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html#access-keys-and-secret-access-keys https://docs.aws.amazon.com/IAM/latest/APIReference/API\_GetAccountSummary.html https://aws.amazon.com/ko/blogs/security/an-easier-way-to-determine-the-presence-of-aws-account-access-keys/ https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 루트 사용자 계정의 MFA 활성화 (1.1.5) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 루트 사용자 계정은 AWS 계정 중 가장 높은 권한이 있는 사용자이다. MFA(Multi-factor Authentication)는 사용자 인증시 사용자 이름과 비밀번호에 추가적인 보안 계층을 제공한다. MFA 활성화시 AWS 웹 사이트에 로그인할 경우 사용자 이름과 비밀번호 인증 후 AWS MFA 장치의 인증 코드를 입력해야 인증이 완료된다. | | |
| 설명 | MFA를 활성화할 경우, 인증 주체가 한정된 시간에만 유효한 값을 입력한 후에 인증이 완료된다. 비밀번호가 손상되더라도 해당 장치를 소유해야 인증이 완료되므로 AWS 웹사이트 접근에 대한 보안이 강화된다. 중요 권한이 부여된 계정의 경우 MFA를 활성화해야 한다. | | |
| 진단 기준 | ▶ 양호 : 루트 계정에 MFA가 설정되어 있는 경우   ▶ 취약 : 루트 계정에 MFA가 설정되어 있지 않은 경우   ▶ Default Value : False | | |
| 조치 방안 | [AWS Console] ※ 루트 사용자 계정의 MFA를 설정하기 위해서는 루트 계정의 자격 증명으로 로그인해야 한다. 다른 자격 증명으로 루트 사용자 계정의 MFA를 설정할 수 없다.  1. https://console.aws.amazon.com/console/ 을 통해 AWS Console에 접속한다. 2. 우측 상단의 계정명을 클릭한다.  3. 드롭다운 메뉴바에서 '내 보안 자격 증명(My Security Credentials)'을 클릭한다.  4. '멀티 팩터 인증(MFA)(Multi-factor authentication (MFA))' 탭을 선택한다. 5. "MFA 활성화(Activate MFA)" 버튼을 클릭한다. 6. "가상 MFA 디바이스(Virtual MFA device)" 을 선택한 후, '계속(Continue)' 버튼을 클릭한다. 7. 앱 설치, QR 코드 스캔, MFA 코드 입력 후 'MFA 할당(Assign MFA)'을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '자격 증명 보고서(Credential Report)'를 클릭한다. 3. '보고서 다운로드(Download Report)'를 클릭한다. 4. csv 파일에서 <root\_account> 의 "mfa\_active" 값이 TRUE 로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 MFA가 활성화되어 있는지 확인한다. 활성화된 경우, "AccountMFAEnabled" 가 1로 표시된다. # aws iam get-account-summary | grep "AccountMFAEnabled" | | |
| Reference | 1.5 Ensure MFA is enabled for the 'root' user account (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_root-user.html#id\_root-user\_manage\_mfa https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_mfa\_enable\_virtual.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 복잡한 비밀번호 규칙 설정(1.1.8) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 최소 비밀번호 길이를 14자 이상 설정하도록 비밀번호 정책을 설정하는 것을 권장한다. 복잡한 비밀번호를 설정하여 무차별 대입 공격을 통해 비밀번호가 노출되지 않도록 해야 한다. | | |
| 설명 | 최소 비밀번호 길이가 14 이상, 최소 한 글자 이상의 대문자 포함, 최소 한 글자 이상의 소문자 포함, 최소 한 글자 이상의 숫자 포함, 최소 한 글자 이상의 특수문자가 포함되도록 복잡한 비밀번호 정책을 설정하는 것을 권장한다. 복잡한 비밀번호를 설정하여 무차별 대입 공격을 통해 비밀번호가 노출되지 않도록 해야 한다. | | |
| 진단 기준 | ▶ 양호 : 최소 비밀번호 길이가 14자 이상, 최소 한 글자 이상의 대문자 포함, 최소 한 글자 이상의 소문자 포함, 최소 한 글자 이상의 숫자 포함, 최소 한 글자 이상의 특수문자가 포함으로 설정되어 있는 경우   ▶ 취약 : 최소 비밀번호 길이 적용이 14자 미만으로 설정되어 있거나 대문자, 소문자, 숫자, 특수문자가 각각 1개 이상 포함되도록 설정되어 있지 않은 경우   ▶ Default Value : 최소 암호 길이 8자, 대문자/소문자/숫자/특수문자 각각 1개 이상 포함되도록 설정되어 있지 않음 | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '계정 설정(Account settings)'을 클릭한다. 3. '암호 정책 변경(Change password policy)' 버튼을 클릭한다. 4. 패스워드 복잡도를 사내 보안 규정에 맞게 입력한다. 5. '변경 내용을 저장합니다(Save changes)'를 클릭한다.  [AWS CLI] 1. 다음 명령어를 실행하여 패스워드 복잡도를 사내 보안 규정에 맞도록 설정한다.  ex) "MinimumPasswordLength" 항목을 14 이상으로 설정하는 것은 다음과 같다. # aws iam update-account-password-policy --minimum-password-length 14  ※ 1개 이상의 라틴 알파벳 대문자(A-Z) 필수, 1개 이상의 라틴 알파벳 소문자(a-z) 필수, 1개 이상의 숫자 필수, 영숫자를 제외한 문자 1개 이상 필수 (!@#$%^&\*()\_+-=[]{}|')을 설정하여 복잡한 패스워드를 설정하도록 유도하는 것을 권장한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '계정 설정(Account settings)'을 클릭한다. 3. 비밀번호 정책(Password policy)에서 패스워드 복잡도가 사내 보안 규정에 맞도록 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 패스워드 복잡도가 사내 보안 규정에 맞도록 설정되어 있는지 확인한다. # aws iam get-account-password-policy | | |
| Reference | 1.8 Ensure IAM password policy requires minimum length of 14 or greater (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_passwords\_account-policy.html https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#configure-strong-password-policy https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 비밀번호 재사용 제한(1.1.9) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | IAM 암호 정책은 동일한 사용자가 지정된 암호를 재사용하지 못하도록 할 수 있다. 암호 정책 설정시 암호 재사용을 방지하도록 설정하는 것을 권장한다. | | |
| 설명 | 암호 재사용 제한을 설정할 경우, 무차별 대입 공격에 대한 계정 복원력이 향상된다. | | |
| 진단 기준 | ▶ 양호 : 암호 재사용 제한(Prevent password reuse) 항목이 설정되어 있는 경우   ▶ 취약 : 암호 재사용 제한 항목이 설정되어 있지 않은 경우   ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '계정 설정(Account settings)'을 클릭한다. 3. '암호 정책 변경(Change password policy)' 버튼을 클릭한다. 4. "암호 재사용 제한(Prevent password reuse)"을 활성화한 후, "기억 n개의 암호(Remember n password(s)" 의 n을 24로 설정한다.  5. '변경 내용을 저장합니다(Save changes)'를 클릭한다.  [AWS CLI] 1. 다음 명령어를 실행하여 "aws iam update-account-password-policy" 항목을 24로 설정한다.  # aws iam update-account-password-policy --password-reuse-prevention 24 | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '계정 설정(Account settings)'을 클릭한다. 3. 비밀번호 정책(Password policy)에서 "마지막 n개 암호를 기억하고 재사용 방지(Prevent password reuse)"가 설정되어 있는지 확인한다. (n이 1이상인지 확인)  [AWS CLI] 1. 다음 명령어를 실행하여 "PasswordReusePrevention" 항목이 1 이상 24 이하로 설정되어 있는지 확인한다.  # aws iam get-account-password-policy | | |
| Reference | 1.9 Ensure IAM password policy prevents password reuse (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_passwords\_account-policy.html https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#configure-strong-password-policy | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 모든 사용자 계정 MFA 활성화(1.1.10) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | MFA(Multi-factor Authentication)는 사용자 인증시 사용자 이름과 비밀번호에 추가적인 보안 계층을 제공한다. MFA 활성화시 AWS 웹 사이트에 로그인할 경우 사용자 이름과 비밀번호 인증 후, AWS MFA 장치의 인증 코드를 입력해야 인증이 완료된다. | | |
| 설명 | MFA 활성화 시 인증 주체가 한정된 시간에만 유효한 값을 입력 후에 인증이 완료된다. 비밀번호가 손상되더라도 해당 장치를 소유해야 인증이 완료되므로 AWS 웹사이트 접근에 대한 보안이 강화된다. 모든 사용자 계정은 되도록 MFA를 활성화해야 한다. | | |
| 진단 기준 | ▶ 양호 : password\_enabled 와 mfa\_active가 "TRUE" 로 설정되어 있는 경우   ▶ 취약 : password\_enabled 와 mfa\_active가 "FALSE" 로 설정되어 있는 경우   ▶ Default Value : FALSE | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. MFA가 활성화되지 않은 각 사용자를 클릭한다.  4. 보안 자격 증명(Security credentials) 탭을 클릭한다.  5. 할당된 MFA 디바이스(Assigned MFA device) 항목 우측의 '관리(Manage)'를 클릭한다.  6. "가상 MFA 디바이스(Virtual MFA device)" 을 선택한 후, '계속(Continue)' 버튼을 클릭한다. 7. 앱 설치, QR 코드 스캔, MFA 코드 입력 후 'MFA 할당(Assign MFA)'을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '자격 증명 보고서(Credential Report)'를 클릭한다. 3. '보고서 다운로드(Download Report)'를 클릭한다. 4. csv 파일에서 사용자들의 "password\_enabled" 와 "mfa\_active" 가 TRUE 로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 자격 증명 보고서를 생성한다. # aws iam generate-credential-report   2. 사용자들 중 "password\_enabled" 와 "mfa\_active" 항목이 모두 true로 설정되어 있는지 확인한다. # aws iam get-credential-report --query 'Content' --output text | base64 -d | cut -d, -f1,4,8 | | |
| Reference | 1.10 Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_mfa.html https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#enable-mfa-for-privileged-users https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_mfa\_enable\_virtual.html https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 초기 액세스 키 사용 제한(1.1.11) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | IAM 사용자 생성시 액세스 키가 발급된다. 초기 생성한 액세스 키가 90일 이후에도 존재할 경우 교체하거나 해당 키를 삭제하는 것을 권장한다. | | |
| 설명 | 초기 설정시 생성되는 액세스 키를 사용하지 않는 경우 삭제하는 것을 권장한다. 되도록 생성 이후 90일 이내 교체해야 한다. 시스템 상 교체 없이 사용해야 하는 경우 해당 리소스 ID에 대해 예외 처리하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 키 생성 시간이 90일 이내이거나 password\_enabled 와 Access key age 둘 다 N/A로 설정되어 있거나, 사용자의 생성 시간과 액세스 키의 생성 완료 시간이 다른 경우   ▶ 취약 : 키 생성 시간이 90일 이후이며 사용자의 생성 시간과 액세스 키의 생성 완료 시간이 같은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 선택한다. 4. 보안 자격 증명(Security credentials) 탭을 클릭한다.  5. 사용자의 '생성 시간(Creation time)'과 액세스 키의 '생성 완료(Created)' 시간이 일치할 경우, 해당 키 우측의 X(삭제) 버튼을 클릭한다.  ※ 관리자 계정도 일반 사용자 계정과 같이 삭제해야 한다.  [AWS CLI] 1. 다음 명령어를 실행하여 액세스 키를 삭제한다.  # aws iam delete-access-key --access-key-id <access-key-id-listed> --user-name <users-name> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 사용자 중 액세스 키 수명(Access key age)과 비밀번호 수명(Password age)가 "없음(None)" 으로 설정된 계정이 없는지 확인한다. 4. 해당 계정을 클릭한다.  5. 보안 자격 증명(Security credentials) 탭을 클릭한다.  6. 사용자의 '생성 시간(Creation time)'과 액세스 키의 '생성 완료(Created)' 시간을 비교한다. 시간이 일치할 경우, 해당 키는 사용자 생성시 만든 액세스 키이다.  ※ '마지막 사용(Last used)' 날짜가 표시되지 않은 액세스 키는 즉시 비활성화 및 삭제해야 한다.   [AWS CLI] 1. 다음 명령어를 실행하여 자격 증명 보고서를 생성한다. # aws iam generate-credential-report  2. 사용자들 중 "password\_enabled" 가 true로 설정되고 "access\_key\_last\_used\_date" 가 N/A로 설정되어 있는지 확인한다.  # aws iam get-credential-report --query 'Content' --output text | base64 -d | cut -d, -f1,4,9,11,14,16 | | |
| Reference | 1.11 Do not setup access keys during initial user setup for all IAM users that have a console password (Manual) | | |
| Site | https://docs.aws.amazon.com/cli/latest/reference/iam/delete-access-key.html https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_users\_create.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 90일 이상 사용하지 않는 자격 증명 비활성화(1.1.12) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | IAM 사용자는 자격 증명(비밀번호, 액세스 키)으로 AWS 리소스에 접근할 수 있다. 90일 이상 사용하지 않은 자격 증명을 삭제하거나 비활성화하는 것을 권장한다. | | |
| 설명 | 불필요한 자격 증명을 비활성화하거나 삭제하는 것을 권장한다. 의도하지 않은 사용자가 해당 자격 증명을 이용하여 AWS 리소스에 접근하지 않도록 사전에 방지해야 한다. | | |
| 진단 기준 | ▶ 양호 : 마지막 로그인 날짜(Console last sign-in), 액세스 키 수명(Access key age)이 90일 미만이고, 마지막으로 사용된 액세스 키(Access key last used)가 "없음(None)"으로 설정되어 있지 않은 경우   ▶ 취약 : 마지막 로그인 날짜, 액세스 키 수명이 90일 이상이고, 마지막으로 사용된 액세스 키가 "없음"으로 설정되어 있는 경우   ▶ Default Value : None | | |
| 조치 방안 | [AWS Console] ※ 사용하지 않는 암호 관리 (IAM 사용자 콘솔 액세스) 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 선택한다. 4. 보안 자격 증명(Security Credentials) 탭을 클릭한다. 5. 콘솔 비밀번호(Console password) 우측의 '관리(Manage)'를 클릭한다. 6. 콘솔 액세스(Console access) 항목에서 "비활성화(Disable)"를 선택한 후, 적용(Apply) 버튼을 클릭한다.  ※ 액세스 키 비활성화 또는 삭제 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 선택한다. 4. 보안 자격 증명(Security Credentials) 탭을 클릭한다. 5. 액세스 키(Access keys) 항목에서 생성된 날짜로부터 90일이 지난 액세스 키를 우측의 "비활성화(Make inactive)" 또는 "X(삭제)" 를 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 우측 상단의 설정 아이콘을 클릭한다. 4. 콘솔 마지막 로그인(Console last sign-in), 액세스 키 ID(Access key ID), 마지막으로 사용한 액세스 키(Access key last used) 항목을 체크한 후, 닫기(Close)를 클릭한다. 5. "콘솔 마지막 로그인" 날짜가 현재 날짜로부터 90일 미만인지 확인한다. ※ 콘솔 마지막 로그인 날짜가 "Never"인 경우는 사용자가 한번도 로그인하지 않았음을 의미한다.  6. "액세스 키 수명(Access key age)"이 현재 날짜로부터 90일 미만인지, "마지막으로 사용된 액세스 키"가 "없음(None)" 인지 확인한다. ※ 사용자가 최근 90일간 콘솔에 로그인하지 않았거나 액세스 키가 생성된지 90일을 초과한 경우, 조치사항에 따라 조치를 취해야 한다.  [AWS CLI] 1. 다음 명령어를 실행하여 자격 증명 보고서를 생성한다. # aws iam generate-credential-report   2. 사용되지 않는 자격 증명이 있는지 확인한다.  # aws iam get-credential-report --query 'Content' --output text | base64 -d | cut -d, -f1,4,5,6,9,10,11,14,15,16  2-1. "password\_enabled" 항목이 TRUE 인 사용자의 경우, "password\_last\_used\_date" 항목이 현재 날짜로부터 90일 미만인지 확인한다. "password\_enabled"가 TRUE 이고 "password\_last\_used"가 No\_Information 으로 설정된 경우, "password\_last\_changed"가 90일 미만인지 확인한다.  2-2. "access\_key\_1\_active" 또는 "access\_key\_2\_active"가 TRUE 인 사용자의 경우, "access\_key\_n\_last\_used\_date"가 90일 미만인지 확인한다. "access\_key\_x\_active(x는 1 또는 2)"가 TRUE 이고 "access\_key\_x\_last\_used\_date"가 N/A 로 설정된 경우, "access\_key\_x\_last\_rotated"가 90일 미만인지 확인한다. | | |
| Reference | 1.12 Ensure credentials unused for 90 days or greater are disabled (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#remove-credentials https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_finding-unused.html https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_passwords\_admin-change-user.html https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_access-keys.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 액세스 키 개수 제한 (1.1.13) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 액세스 키를 사용하여 프로그래밍 방식 요청을 AWS CLI 또는 AWS API(직접 또는 AWS SDK 사용)에 서명할 수 있으므로 액세스 키가 2개일 경우 1개를 삭제 또는 비활성화하는 것을 권장한다. | | |
| 설명 | 액세스 키는 IAM 사용자 또는 AWS 계정 루트 사용자에 대한 장기 자격 증명이다. 액세스 키를 사용하여 프로그래밍 방식 요청을 AWS CLI 또는 AWS API에 서명할 수 있다. 계정을 보호하는 가장 좋은 방법 중 하나는 사용자가 여러 개의 액세스 키를 가지지 못하도록 하는 것이다. | | |
| 진단 기준 | ▶ 양호 : 활성화 된 액세스 키가 1개인 경우   ▶ 취약 : 활성화 된 액세스 키가 2개인 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 선택한다. 4. 보안 자격 증명(Security credentials) 탭을 클릭한다.  5. 액세스 키 중 생성된지 90일 미만의 키 1개 만을 남겨놓고 나머지 키는 비활성화(Make inactive) 또는 삭제한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. "액세스 키 수명(Access key age)"이 존재하는 사용자를 클릭한다. 4. 보안 자격 증명(Security credentials) 탭을 클릭한다.  5. 활성화 된 액세스 키가 1개 인지 확인한다.  [AWS CLI] 1. 모든 사용자 목록을 확인한다.  # aws iam list-users --query "Users[\*].UserName"   2. 다음 명령어를 실행하여 IAM 사용자와 연결된 액세스 키의 현재 상태(Status)가 "Active" 인지 확인한다. 결과값에 metadata("Username", "AccessKeyId", "Status", "CreateDate")가 출력되어야 한다.  # aws iam list-access-keys --user-name <user-name>  ※ 둘 이상의 액세스 키의 현재 상태가 "Active" 인 경우는 권장하지 않는다. | | |
| Reference | 1.13 Ensure there is only one active access key available for any single IAM user (Automated) | | |
| Site | https://docs.aws.amazon.com/general/latest/gr/aws-access-keys-best-practices.html https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_access-keys.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 액세스 키 90일 이내 교체(1.1.14) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 액세스 키는 액세스 키 ID와 비밀 키로 구성된다. AWS 사용자는 자체 액세스 키가 있을 경우에만 AWS CLI, AWS SDK 사용 및 AWS 서비스용 API를 사용하여 직접 HTTP를 호출할 수 있다. 모든 액세스 키는 정기적으로(90일 이내) 교체하는 것을 권장한다. | | |
| 설명 | 주기적으로 액세스 키를 교체할 경우 키가 손상되거나 사용하지 않는 계정과 관련된 액세스 키를 사용할 가능성을 줄일 수 있다. 액세스 키가 손상, 분실, 도난되었을 경우 이전 키를 사용하여 특정 데이터에 접근할 수 없도록 정기적으로 교체할 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 액세스 키 수명(Access key age)이 90일 이내인 경우   ▶ 취약 : 액세스 키 수명이 90일 이상인 경우   ▶ Default Value : None | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 선택한다. 4. 보안 자격 증명(Security credentials) 탭을 클릭한다.  5. 생성된지 90일이 지난 액세스 키 우측의 'X' 버튼을 클릭하여 해당 키를 삭제한다.  6. '액세스 키 만들기(Create access key)' 버튼을 클릭하여 새로운 액세스 키를 생성한다.  [AWS CLI] 1. 사용자의 액세스 키 목록을 출력한다.  # aws iam list-access-keys --user-name <user\_name>  2. 첫번째 액세스 키가 활성화되어 있는 상태에서 두번째 액세스 키를 생성한다. # aws iam create-access-key --user-name <user\_name>  3. 모든 애플리케이션과 도구에서 새 액세스 키를 사용하도록 설정한다.  4. 며칠이 지난 이후, 첫번째 액세스 키가 여전히 사용되고 있는지 확인한다.  # aws iam get-access-key-last-used --access-key-id <first\_access\_key\_id>  5. 이전 키가 사용되고 있지 않더라도 해당 키를 즉시 삭제하는 것보다 비활성화 하는 것을 권장한다.  # aws iam update-access-key --user-name <user\_name> --access-key-id <first\_access\_key\_id> --status Inactive  6. 새 액세스 키만 사용하여 애플리케이션이 정상 작동하는지 확인한다.   7. 모든 애플리케이션과 도구에서 새 액세스 키를 사용하도록 설정된 경우, 첫번째 액세스 키를 삭제한다.  # aws iam delete-access-key --user-name <user\_name> --access-key-id <first\_access\_key\_id> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 우측 상단의 설정 아이콘을 클릭한다. 4. "액세스 키 수명(Access key age)"이 현재 날짜로부터 90일 미만인지, "마지막으로 사용된 액세스 키(Access key last used)"가 "없음(None)" 인지 확인한다. ※ 사용자가 최근 90일간 콘솔에 로그인하지 않았거나 액세스 키가 생성된지 90일을 초과한 경우, 조치사항에 따라 조치를 취해야 한다.  [AWS CLI] 1. 다음 명령어를 실행하여 자격 증명 보고서를 생성한다. # aws iam generate-credential-report   2. "access\_key\_1\_last\_rotated" 가 현재 날짜로부터 90일 미만인지 확인한다. "access\_key\_1\_last\_rotated" 가 None 인 경우 액세스 키가 없다는 것을 의미한다. # aws iam get-credential-report --query 'Content' --output text | base64 -d | cut -d, -f1,10 | | |
| Reference | 1.14 Ensure access keys are rotated every 90 days or less (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#rotate-credentials https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_finding-unused.html https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html#access-keys-and-secret-access-keys https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_access-keys.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | IAM 정책 설정 시 그룹으로 권한 관리(1.1.15) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 사용자 권한 관리 및 과도한 권한 할당을 제한하기 위하여 그룹을 통해 IAM 정책을 할당하는 것을 권장한다. | | |
| 설명 | IAM 사용자는 IAM 정책을 통해 서비스, 기능 및 데이터에 접근할 수 있다. 사용자에 대한 정책을 설정하는 방법은 3가지가 있다.  1) 사용자 정책을 직접 설정(인라인 정책)  2) 사용자에 정책을 직접 연결  3) 정책이 붙여진 IAM 그룹에 사용자를 추가  ※ 이 3가지 중 3번째 방식을 권장한다.  그룹을 통해 IAM 정책을 할당할 경우, 조직의 역할을 일괄적으로 관리할 수 있어 유지보수 측면에서 유리하다. 또한, 과도한 권한 할당의 가능성을 줄일 수 있다. 인라인 형태로 관리하거나 각 사용자마다 정책을 할당할 경우, 관리 측면에서 어려우며 특정 작업 이후 해제해야 하는 권한을 누락할 가능성이 생긴다. | | |
| 진단 기준 | ▶ 양호 : 그룹으로 권한 관리를 하는 경우   ▶ 취약 : 직접 연결로 권한 관리를 하는 경우    ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 사용자에게 직접 정책이 연결되어 있는 경우, 아래 조치에 따라 직접 연결된 권한을 삭제하고 그룹을 통해 권한 관리를 하도록 설정한다.  1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 선택한다. 4. 권한(Permissions) 탭을 클릭한다.  5. "Permissions Policies" 에서 직접 연결된 권한을 삭제한다. ※ 관리자를 제외한 일반 사용자일 경우, 그룹으로 권한을 할당하는 것을 권장한다 6. '권한 추가(Add permissions)' 버튼을 클릭한다.  7. 해당 그룹이 필요한 권한만 설정한 그룹을 선택한 후, '다음: 검토(Next: Review)'를 클릭한다.  8. '권한 추가(Add permissions)'를 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 각 사용자를 클릭한다. 4. 권한(Permissions) 탭을 클릭한다.  5. "Permissions Policies" 에서 직접 연결된 권한이 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 모든 사용자를 출력한다.  # aws iam list-users --query 'Users[\*].UserName' --output text   2. 각 사용자에 대해 연결된 정책이 존재하는지 확인한다.  # aws iam list-attached-user-policies --user-name <iam\_user>  # aws iam list-user-policies --user-name <iam\_user>   3. 연결된 정책이 존재하는 경우, 해당 사용자가 인라인 정책 또는 직접 정책을 갖고 있음을 의미한다. | | |
| Reference | 1.15 Ensure IAM Users Receive Permissions Only Through Groups (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html https://docs.aws.amazon.com/IAM/latest/UserGuide/access\_policies\_managed-vs-inline.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | "\*:\*" administrative 권한 제거(1.1.16) | 위험도 | High |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 최소 권한 원칙을 위해 사용자가 수행하는 작업에 필요한 권한만을 부여하는 것을 권장한다. | | |
| 설명 | 과도한의 권한을 설정한 이후 권한을 줄이는 방식보다 처음 정책을 설정하는 단계에서부터 최소한의 권한으로 시작하고 필요에 따라 추가 사용 권한을 부여하는 것이 더 안전하다. 사용자가 수행해야 하는 데 필요한 최소한의 권한 집합으로 제한하지 않고 전체 관리 권한을 연결할 경우, 의도하지 않은 작업에 리소스가 노출된다. "Resource": "\*", "Action": "\*", "Effect": "Allow" 라는 구문이 있는 IAM 정책은 제거해야 한다. | | |
| 진단 기준 | ▶ 양호 : "\*:\*" administrative 권한이 할당되어 있지 않은 경우   ▶ 취약 : "\*:\*" administrative 권한이 할당되어 있는 경우   ▶ Default Value : None ※ "\*:\*" administrative 권한이 필요한 경우 관리자의 승인을 받아야 한다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후, '분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 해당 정책에서 모든 IAM 사용자를 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 해당 정책에서 모든 IAM 그룹을 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 해당 정책에서 모든 IAM 역할을 분리시킨다.  # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 IAM 정책 목록를 출력한다.  # aws iam list-policies --only-attached --output text   2. 반환된 각 정책에 대해 모든 관리(full administrative) 권한을 허용하는 정책이 있는지 확인한다.  # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version>   3. 출력된 정책 중 "Effect": "Allow", "Action": "\*", "Resource": "\*" 으로 설정이 된 구문이 없는지 확인한다. | | |
| Reference | 1.16 Ensure IAM policies that allow full "\*:\*" administrative privileges are not attached (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html https://docs.aws.amazon.com/IAM/latest/UserGuide/access\_policies\_managed-vs-inline.html https://docs.aws.amazon.com/cli/latest/reference/iam/index.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | IAM 역할을 통한 리소스 액세스 제어(1.1.18) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | IAM 역할을 통해 AWS 리소스에 대한 액세스를 제어하는 것을 권장한다. | | |
| 설명 | AWS 인스턴스 내에서 AWS 접근 시 AWS 키를 통해 AWS 호출로 인코딩하거나 필요한 접근에 대한 적절한 권한 정책이 있는 역할에 인스턴스를 할당하여 수행할 수 있다. "AWS Access"는 AWS 리소스에 접근하거나 AWS 계정 리소스를 관리하기 위해 AWS의 API에 접근하는 것을 의미한다. AWS IAM 역할은 AWS 자체적으로 외부에서 사용할 수 있는 자격 증명 공유 및 교체와 관련된 위협을 줄인다. 자격 증명이 손상된 경우 접근 권한을 부여한 AWS 계정 외부에서 자격 증명을 사용할 수 있다. 해당 자격 증명을 주기적으로 교체해야 하는데, 자격 증명이 컴파일 된 애플리케이션으로 인코딩된 경우 자격 증명을 교체하기 어렵다. 반면 역할을 통해 접근 제어를 할 경우, 공격자가 특정 인스턴스에 대한 접근 권한을 얻은 이후 이를 유지하여 해당 인스턴스에 연결된 권한을 사용해야 하므로 보안 계층이 더 높으며 유지보수 측면에서 유리하다. | | |
| 진단 기준 | ▶ 양호 : 인스턴스에 최소한의 권한만 적용된 IAM 역할이 할당되어 있는 경우   ▶ 취약 : 인스턴스에 최소한의 권한만 적용된 IAM 역할이 할당되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | ※ 기존 인스턴스에 IAM 역할을 바로 할당하는 방법은 현재 AWS에서 지원하지 않으며, IAM 역할은 인스턴스를 시작할 때만 연결할 수 있다. 인스턴스에 IAM 역할을 추가하기 위해서는 새 인스턴스를 생성해야 한다.  ※ 인스턴스가 개인 IP에 대한 외부 종속성이 없는 경우, 공용 주소가 탄력 IP인 경우에 아래와 같은 조치를 시행한다.   [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 좌측 사이드 바에서 '역할(Roles)' 을 클릭한다.  3. 새 역할을 생성한다.  4. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 5. 기존 인스턴스와 설정이 동일한 새로운 인스턴스를 새로 생성한 역할을 선택하여 생성한다.  6. 기존 인스턴스와 새 인스턴스를 모두 종료한다.  7. 두 인스턴스의 디스크를 분리한다, . 8. 기존 인스턴스의 디스크를 새 인스턴스에 연결하여 부팅한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 인스턴스 요약(Instance summary)에서 인스턴스 프로파일 ARN(Instance Profile ARN) 또는 IAM 역할(IAM Role)이 존재하는지 확인한다.  ※ 존재하지 않는 경우, 인스턴스에 역할이 할당되어 있지 않다는 것을 의미한다.  5. 역할이 있는 경우 해당 역할을 선택한 후, 최소한의 권한이 할당되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 인스턴스에 할당된 "IamInstanceProfile"이 존재하는지 확인한다.  # aws ec2 describe-instances --instance-ids <instance id> | | |
| Reference | 1.18 Ensure IAM instance roles are used for AWS resource access from instances (Manual) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_roles\_use\_switch-role-ec2.html https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 만료된 SSL/TLS 인증서 삭제(1.1.19) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 만료된 인증서는 애플리케이션, 웹 사이트의 신뢰성에 손상을 줄 수 있기 때문에 즉시 삭제하는 것을 권장한다. | | |
| 설명 | AWS에서 애플리케이션 또는 웹 사이트에 HTTPS 연결을 사용하기 위해서 SSL/TLS 서버 인증서가 필요하다. ACM(Amazon Certificate Manager) 또는 IAM을 통해 서버 인증서를 저장하고 배포할 수 있다. ACM이 지원하지 않은 리전에서 HTTPS 연결을 사용해야 할 때만 IAM을 인증서 관리자로 사용하는 것을 권장하며, IAM은 모든 리전에서 서버 인증서 배포를 지원하지만 AWS에서 사용하려면 외부 공급자로부터 인증서를 받아야 한다.  만료된 SSL/TLS 인증서는 애플리케이션, 웹 사이트의 신뢰성에 손상을 줄 수 있기 때문에 즉시 삭제하는 것을 권장한다. 만료된 인증서를 삭제할 경우, ELB(Elastic Load Balancer)와 같은 리소스에 잘못된 인증서가 배포될 위험을 줄일 수 있다. 만료된 인증서 삭제는 애플리케이션에 영향을 미칠 수 있기 때문에 이를 고려하여 수행해야 한다. | | |
| 진단 기준 | ▶ 양호 : 만료된 SSL/TLS 인증서가 없는 경우   ▶ 취약 : 만료된 SSL/TLS 인증서가 있는 경우   ▶ Default Value : 만료된 인증서는 삭제되지 않음 | | |
| 조치 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 만료된 인증서를 삭제한다. # aws iam delete-server-certificate --server-certificate-name <certificate\_name> | | |
| 감사 방안 | ※ 현재 인증서 만료 정보는 AWS Console에서는 확인할 수 없다. AWS API를 통해 IAM에 저장된 SSL/TLS 인증서에 대한 정보를 확인하고자 하는 경우 CLI를 사용해야 한다.   [AWS CLI]  1. 다음 명령어를 실행하여 IAM에 저장된 서버 인증서를 확인한다.  # aws iam list-server-certificates   출력 결과 예시는 아래와 같다.  {  "ServerCertificateMetadataList": [   {   "ServerCertificateId": "EHDGFRW7EJFYTE88D",  "ServerCertificateName": "MyServerCertificate",  "Expiration": "2018-07-10T23:59:59Z",  "Path": "/",   "Arn": "arn:aws:iam::012345678910:server-certificate/MySSLCertificate",   "UploadDate": "2018-06-10T11:56:08Z"  }   ]  }   출력 값은 IAM에 현재 저장된 모든 SSL/TLS 인증서와 해당 메타 데이터(이름, ID, 만료 날짜 등)가 포함된 배열을 반환한다. 각 SSL/TLS 인증서의 "ServerCertificateName" 및 "Expiratoin(만료 날짜)" 파라미터 값을 확인하고, AWS IAM에 현재 저장된 인증서 중 만료된 인증서가 있는지 확인한다. { { "ServerCertificateMetadataList": [] }가 반환되는 경우 만료된 인증서가 없다는 것을 의미한다 (인증서가 존재하지 않는다는 것이 아니다). | | |
| Reference | 1.19 Ensure that all the expired SSL/TLS certificates stored in AWS IAM are removed (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_server-certs.html https://docs.aws.amazon.com/cli/latest/reference/iam/delete-server-certificate.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | IAM 액세스 분석기 활성화(1.1.20) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | 모든 리전 및 리소스에 대한 IAM 정책에 대해 IAM 액세스 분석기를 사용하는 것을 권장한다. | | |
| 설명 | AWS IAM 액세스 분석기를 사용할 경우 조직 및 계정에서 외부 객체와 공유되는 리소스를 식별할 수 있다. 이를 통해 리소스 및 데이터에 대한 의도하지 않은 접근이 가능한지 확인할 수 있다. 액세스 분석기는 논리 기반 추론을 사용하여 AWS 환경에서 리소스 기반 정책을 분석 및 외부 주체와 공유되는 리소스를 식별한다. AWS IAM 액세스 분석기는 S3 버킷, KMS(Key Management Service) 키, IAM 역할, AWS Lambda, SQS(Simple Queu Services) Queue 등에 대한 모든 정책을 지속적으로 모니터링한다. 리소스가 있는 모든 리전에 액세스 분석기를 생성하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 모든 리전에 IAM 액세스 분석기(Access Analyzer)가 활성화되어 있는 경우   ▶ 취약 : IAM 액세스 분석기가 비활성화되어 있는 경우   ▶ Default Value : inactive | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '액세스 분석기(Access Analyzer)'를 클릭한다. 3. "분석기 생성(Create analyzer)" 버튼을 클릭한다. 4. 정보를 확인한 후, 분석기 생성 버튼을 클릭한다. ※ 리소스가 있는 모든 리전에 액세스 분석기를 생성해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '액세스 분석기(Access Analyzer)'를 클릭한다. 3. 해당 기능이 활성화되어 있는지 확인한다.  ※ 리소스가 있는 모든 리전에 액세스 분석기가 활성화되어 있는지 확인한다. | | |
| Reference | 1.20 Ensure that IAM Access analyzer is enabled for all regions (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/what-is-access-analyzer.html https://docs.aws.amazon.com/IAM/latest/UserGuide/access-analyzer-getting-started.html https://docs.aws.amazon.com/cli/latest/reference/accessanalyzer/get-analyzer.html https://docs.aws.amazon.com/cli/latest/reference/accessanalyzer/create-analyzer.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 90일 이상 미접속 계정 삭제(1.1.22) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | IAM Configuration | | |
| 요약 | IAM 사용자는 자격 증명(비밀번호, 액세스 키)으로 AWS 리소스에 접근할 수 있다. 90일 이상 접속하지 않은 계정은 삭제하거나 비활성화하는 것을 권장한다. | | |
| 설명 | 장기간 접속하지 않은 계정에 대해 삭제 및 비활성화하는 것을 권장한다. 관리되지 않은 계정은 의도하지 않은 사용자가 해당 계정을 악의적으로 사용할 가능성이 있으므로 사용하지 않은 계정, 계약이 만료된 계정, 외부 사용자의 계정 등 업무가 종료되는 대로 삭제하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 마지막 로그인 날짜(Console last sign-in) 또는 액세스 키 수명(Access key age)이 90일 미만이거나 마지막으로 사용된 액세스 키(Access key last used)가 90일 미만이거나 "없음(None)"으로 설정되어 있지 않은 경우   ▶ 취약 : 마지막 로그인 날짜, 액세스 키 수명이 90일 이상이고 마지막으로 사용된 액세스 키가 90일 이상이거나 "없음"으로 설정되어 있는 경우   ▶ Default Value : None | | |
| 조치 방안 | [AWS Console] ※ 사용자를 삭제하기 전에 먼저 사용자의 액세스키를 비활성화 및 삭제해야 한다.  1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 클릭한다.  4. 보안 자격 증명(Security credentials) 탭을 클릭한다.  5. 해당 액세스 키의 "비활성화(Make inactive)"를 클릭한 후, 비활성화(Deactivate) 버튼을 클릭한다.  6. 비활성화 한 키의 "x" 를 클릭하여 해당 키를 삭제한다.  7. 상단의 '사용자 삭제(Delete user)' 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/home# 를 통해 IAM 대시보드(IAM dashboard)에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 우측 상단의 설정 아이콘을 클릭한다. 4. "콘솔 마지막 로그인(Console last sign-in)", "액세스 키 ID(Access key ID)", "마지막으로 사용한 액세스 키(Access key last used)"를 선택한 후 '닫기(Close)'를 클릭한다. 5. "콘솔 마지막 로그인" 날짜가 90일 미만인지 확인한다. 6. "액세스 키 수명(Access key age)" 이 90일 미만인지 그리고 "마지막으로 사용된 액세스 키" 가 "없음(None)" 인지 확인한다. ※ 사용자가 최근 90일 간 콘솔에 로그인하지 않았거나, 액세스 키가 90일을 초과한 경우 조치사항에 따라 조치를 취해야 한다.  [AWS CLI] 1. 다음 명령어를 실행하여 자격 증명 보고서를 생성한다. # aws iam generate-credential-report   2. "password\_enabled" 가 TRUE, "password\_last\_used\_date" 가 90일 미만 인지 확인한다. ("password\_enabled" 가 TRUE 이거나 "password\_last\_used" 가 No\_Information 로 설정된 경우, "password\_last\_changed" 가 90일 미만인지 확인한다) access\_key\_1\_active 와 access\_key\_2\_active가 "TRUE" 로 설정되어 있을 경우, access\_key\_n\_last\_used\_date 가 90일 미만인지 확인한다. (access\_key\_1\_active와 access\_key\_2\_active 가 TRUE 이고 access\_key\_x\_last\_used\_date 가 N/A 로 된 경우 access\_key\_x\_last\_rotated 가 90일 미만인지 확인) # aws iam get-credential-report --query 'Content' --output text | base64 -d | cut -d, -f1,4,5,6,9,10,11,14,15,16 | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#remove-credentials https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_finding-unused.html https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_passwords\_admin-change-user.html https://docs.aws.amazon.com/IAM/latest/UserGuide/id\_credentials\_access-keys.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:CreatePolicyVersion 권한 해제(1.1.23) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:CreatePolicyVersion 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:CreatePolicyVersion 권한은 사용자가 접근할 수 있는 새 버전의 IAM 정책을 만들 수 있는 권한으로, 이를 통해 사용자 정의 권한을 설정할 수 있다. 새 버전의 정책 생성시 해당 버전이 default 버전으로 설정할 수 있는 권한이 포함되어 있다. 따라서 iam:CreatePolicyVersion 권한이 없어도 기본 버전으로 설정할 수 있다는 것을 의미한다. 권한을 상승할 경우, 사용자가 AWS 계정에 대한 전체 관리자 권한을 얻을 수 있게 된다. 따라서 되도록 해당 권한을 할당하지 않는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:CreatePolicyVersion 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:CreatePolicyVersion 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 모든 사용자, 그룹, 역할을 선택한 후, '분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:CreatePolicyVersion",  "iam:SetDefaultPolicyVersion"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text   2. 반환된 각 정책에 대해 iam:CreatePolicyVersion 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/access\_policies\_managed-versioning.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:SetDefaultPolicyVersion 권한 해제(1.1.24) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:SetDefaultPolicyVersion 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:SetDefaultPolicyVersion 권한을 통해 현재 사용 중이거나 기존 정책 버전을 통해 권한을 상승할 수 있다. 접근 권한이 있는 정책에 버전이 여러 개일 경우, 현재 설정된 기존 정책을 다른 정책으로 변경할 수 있다. 따라서 비활성화된 정책에 따라 권한 상승을 할 수 있거나 없을 수 있다. 되도록 해당 권한을 할당하지 않는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:SetDefaultPolicyVersion 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:SetDefaultPolicyVersion 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후, '분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:CreatePolicyVersion",  "iam:SetDefaultPolicyVersion"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text IAM   2. 반환된 각 정책에 대해 iam:SetDefaultPolicyVersion 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, ec2:RunInstances 권한 해제(1.1.25) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole and ec2:RunInstances 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PassRole과 ec2:RunInstances 권한을 갖고 있는 경우, 새로운 EC2 인스턴스를 생성 및 기존 EC2 인스턴스의 프로파일과 서비스 역할을 새로 생성한 인스턴스에 전달할 수 있게 된다. 해당 인스턴스에 로그인하여 EC2 인스턴스 메타데이터에 연결된 AWS 키를 요청하여 해당 인스턴스의 프로파일 및 서비스 역할이 가진 모든 권한에 접근할 수 있게 된다. 해당 공격은 공격자가 인스턴스 프로파일 및 역할에 대한 권한 집합에 접근할 수 있는 권한을 부여하여 권한 상승이 없는 계정, AWS 계정의 전체 관리자 계정 등에 접근할 수도 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole 과 ec2:RunInstances 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole 또는 ec2:RunInstances 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PassRole, ec2:RunInstances 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/id\_roles\_use\_switch-role-ec2.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:CreateAccessKey 권한 해제(1.1.26) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:CreateAccessKey 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:CreateAccessKey 권한을 갖고 있는 경우, AWS 환경에 있는 다른 사용자에게 속한 액세스 키 ID와 secret 액세스 키를 생성할 수 있다(해당 사용자가 액세스 키를 2개 갖고 있지 않은 경우에만 해당). 공격자는 액세스 키를 생성할 수 있는 모든 사용자와 동일한 수준의 권한을 갖게 되며, 권한 상승이 없는 사용자 및 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:CreateAccessKey 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:CreateAccessKey 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:CreateAccessKey"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text   2. 반환된 각 정책에 대해 iam:PassRole, ec2:RunInstances 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/access\_permissions-required.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:CreateLoginProfile 권한 해제(1.1.27) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:CreateLoginProfile 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:CreateLoginProfile 권한을 갖고 있는 경우, 로그인 프로파일이 설정되어 있지 않은 사용자의 AWS 콘솔에 로그인하는 데 사용되는 비밀번호를 생성할 수 있다. 공격자는 로그인 프로파일을 생성할 수 있는 모든 사용자와 동일한 수준의 권한을 갖게 되며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:CreateLoginProfile 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:CreateLoginProfile 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:CreateLoginProfile"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:CreateLoginProfile 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:UpdateLoginProfile 권한 해제(1.1.28) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:UpdateLoginProfile 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:UpdateLoginProfile 권한을 갖고 있는 경우, 로그인 프로파일이 이미 설정되어 있는 모든 사용자의 AWS 콘솔에 로그인하는데 사용되는 암호를 변경할 수 있다. 공격자는 로그인 프로파일을 업데이트할 수 있는 모든 사용자와 동일한 수준의 권한을 갖게 되며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:UpdateLoginProfile 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:UpdateLoginProfile 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:UpdateLoginProfile"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:UpdateLoginProfile 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:AttachUserPolicy 권한 해제(1.1.29) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:AttachUserPolicy permission 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:AttachUserPolicy 권한을 갖고 있는 경우 권한 상승을 할 수 있는 사용자에 정책을 연결하여 해당 정책의 권한을 공격자에게 추가하여 권한을 상승할 수 있다. 공격자는 해당 방법을 사용하여 AdministratorAccess AWS 관리 정책을 권한 상승할 수 있는 사용자에게 연결 및 부여할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:AttachUserPolicy 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:AttachUserPolicy 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:AttachUserPolicy"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:AttachUserPolicy 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:AttachGroupPolicy 권한 해제(1.1.30) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:AttachGroupPolicy 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:AttachGroupPolicy 권한을 갖고 있는 경우, 권한 상승을 할 수 있는 그룹에 정책을 연결하여 해당 정책의 권한을 공격자에게 추가하여 권한 상승할 수 있다. 공격자는 해당 방법을 사용하여 AdministratorAccess AWS 관리 정책을 권한 상승할 수 있는 그룹에 연결 및 부여할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:AttachGroupPolicy 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:AttachGroupPolicy 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:AttachGroupPolicy"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:AttachGroupPolicy 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:AttachRolePolicy, sts:AssumeRole 권한 해제(1.1.31) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:AttachRolePolicy 와 sts:AssumeRole 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:AttachGroupPolicy 와 sts:AssumeRole 권한을 갖고 있는 경우, 권한 상승을 할 수 있는 역할에 정책을 연결하여 해당 정책의 권한을 공격자에게 추가하여 권한 상승할 수 있다. 공격자는 해당 방법을 사용하여 AdministratorAccess AWS 관리 정책을 권한 상승할 수 있는 역할에 연결 및 부여할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:AttachRolePolicy 와 sts:AssumeRole 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:AttachRolePolicy 또는 sts:AssumeRole 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:AttachRolePolicy, sts:AssumeRole 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PutUserPolicy 권한 해제(1.1.32) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PutUserPolicy 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PutUserPolicy 권한을 갖고 있는 경우, 사용자에 기존 설정된 인라인 정책을 업데이트하고 업데이트된 정책을 공격자에 권한을 추가하여 권한 상승할 수 있다. 공격자는 모든 리소스에 작업을 수행할 수 있는 권한을 부여하는 정책을 지정할 수 있어 전체 관리자 권한을 부여할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 권한이 iam:PutUserPolicy 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 권한이 iam:PutUserPolicy 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:PutUserPolicy"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PutUserPolicy 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PutGroupPolicy 권한 해제(1.1.33) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PutGroupPolicy 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PutGroupPolicy 권한을 갖고 있는 경우, 그룹에 설정된 기존 인라인 정책을 업데이트하고 업데이트된 정책을 공격자에 권한을 추가하여 권한 상승할 수 있다. 공격자는 모든 리소스에 작업을 수행할 수 있는 권한을 부여하는 정책을 지정할 수 있어 전체 관리자 권한을 부여할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PutGroupPolicy 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PutGroupPolicy 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:PutGroupPolicy"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PutGroupPolicy 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PutRolePolicy, sts:AssumeRole 권한 해제(1.1.34) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PutRolePolicy 와 sts:AssumeRole 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PutRolePolicy 와 sts:AssumeRole 권한을 갖고 있는 경우, 역할에 설정된 기존 인라인 정책을 업데이트하고 업데이트된 정책을 공격자에 권한을 추가하여 권한 상승할 수 있다. 공격자는 모든 리소스에 작업을 수행할 수 있는 권한을 부여하는 정책을 지정할 수 있어 전체 관리자 권한을 부여할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PutRolePolicy 와 sts:AssumeRole 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PutRolePolicy 또는 sts:AssumeRole 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PutRolePolicy, sts:AssumeRole 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:AddUserToGroup 권한 해제(1.1.35) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:AddUserToGroup 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:AddUserToGroup 권한을 갖고 있는 경우, AWS 계정 안에 존재하는 IAM 그룹에 자신을 추가할 수 있다. 공격자는 해당 그룹을 통해 계정에 존재하는 모든 그룹에 대한 권한을 얻을 수 있으므로 권한 상승할 수 있는 full administrator access 권한 또한 얻을 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:AddUserToGroup 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:AddUserToGroup 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  또는 정책을 다음과 같이 설정한다. {  "Version": "2012-10-17",  "Statement": [  {  "Effect": "Deny",  "Action": [  "iam:AddUserToGroup"  ],  "Resource": "arn:aws:iam::\*:policy/POLICY-NAME"  }  ] }  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:AddUserToGroup 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:UpdateAssumeRolePolicy, sts:AssumeRole 권한 해제(1.1.36) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:UpdateAssumeRolePolicy 와 sts:AssumeRole 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:UpdateAssumeRolePolicy 와 sts:AssumeRole 권한을 갖고 있는 경우, 기존 assume 역할 정책 문서를 변경하여 해당 역할을 할당하도록 할 수 있다. 공격자는 계정의 모든 역할에 연결된 권한을 얻을 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:UpdateAssumeRolePolicy 와 sts:AssumeRole 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:UpdateAssumeRolePolicy 또는 sts:AssumeRole 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:UpdateAssumeRolePolicy, sts:AssumeRole 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, lambda:CreateFunction, lambda:InvokeFunction 권한 해제(1.1.37) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:InvokeFunction 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PassRole, lambda:CreateFunction, lambda:InvokeFunction 권한은 기존 IAM 역할을 SDK로 AWS 라이브러리를 가져오는 코드가 포함된 새로운 Lambda 함수에 전달하고, 해당 기능을 통해 선택한 작업을 수행하는 것으로 권한 상승할 수 있다. AWS API를 통해 함수를 호출하고 해당 코드를 실행할 수 있다. 이 경우 계정에 있는 Lambda 서비스 역할과 연결된 권한에 대한 액세스 권한이 사용자에게 부여되어 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:InvokeFunction 권한이 모두 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:InvokeFunction 권한 중 하나라도 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PassRole, lambda:CreateFunction, lambda:InvokeFunction 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege  https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping, dynamodb:CreateTable, dynamodb:PutItem 권한 해제(1.1.38) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping, dynamodb:CreateTable, dynamodb:PutItem 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | lambda:InvokeFunction 권한이 없으며 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping, dynamodb:CreateTable, dynamodb:PutItem 권한이 있는 사용자는 기존 IAM 역학을 선택한 프로그래밍 언어로 관련 AWS 라이브러리를 가져오는 코드가 포함된 새로운 Lambda 함수에 전달하여 해당 기능을 사용하여 원하는 작업을 수행함으로 권한 상승할 수 있다. 공격자는 계정에 있는 Lambda 서비스 역할과 관련한 권한에 액세스할 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping, dynamodb:CreateTable, dynamodb:PutItem 권한이 모두 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping, dynamodb:CreateTable, dynamodb:PutItem 권한 중 하나라도 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping, dynamodb:CreateTable, dynamodb:PutItem 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping 권한 해제(1.1.39) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | lambda:InvokeFunction 권한이 없으며 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping 권한이 있는 사용자는 기존 IAM 역학을 선택한 프로그래밍 언어로 관련 AWS 라이브러리를 가져오는 코드가 포함된 새로운 Lambda 함수에 전달하여 해당 기능을 사용하여 원하는 작업을 수행함으로 권한 상승할 수 있다. 공격자는 계정에 있는 Lambda 서비스 역할과 관련한 권한에 액세스할 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping 권한이 모두 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping 권한 중 하나라도 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PassRole, lambda:CreateFunction, lambda:CreateEventSourceMapping 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, glue:CreateDevEndpoint 권한 해제(1.1.40) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole 와 glue:CreateDevEndpoint 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PassRole, glue:CreateDevEndpoint 권한을 갖고 있는 경우, 새로운 AWS Glue 개발 엔드포인트를 생성 및 기존 서비스 역할을 해당 엔드포인트에 전달할 수 있다. 이후 인스턴스에 SSH로 원격 접속 및 AWS CLI를 통해 역할이 액세스할 수 있는 권한에 접근한다. 공격자는 계정에 있는 Glue와 연관된 권한에 액세스할 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole 와 glue:CreateDevEndpoint 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole 또는 glue:CreateDevEndpoint 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PassRole, glue:CreateDevEndpoint 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | glue:UpdateDevEndpoint 권한 해제(1.1.41) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 glue:UpdateDevEndpoint 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | glue:UpdateDevEndpoint 권한을 갖고 있는 경우, Glue 개발 엔드포인트에 연관된 SSH 공개키를 업데이트한 후 원격 접속할 수 있다. 공격자는 연결된 역할이 액세스할 수 있는 권한에 액세스할 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 glue:UpdateDevEndpoint 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 glue:UpdateDevEndpoint 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 glue:UpdateDevEndpoint 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, cloudformation:CreateStack 권한 해제(1.1.42) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole 와 cloudformation:CreateStack 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PassRole 와 cloudformation:CreateStack 권한을 갖고 있는 경우, CloudFormation 스택을 생성할 때 전달된 역할의 권한을 사용하여 작업을 수행 및 리소스를 생성하는 CloudFormation 템플릿을 생성하여 권한 상승할 수 있다. 공격자는 CloudFormation 스택을 만들 때 전달된 역할과 관련된 권한에 액세스할 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole 와 cloudformation:CreateStack 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole 또는 cloudformation:CreateStack 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 glue:UpdateDevEndpoint 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | iam:PassRole, datapipeline:CreatePipeline, datapipeline:PutPipelineDefinition 권한 해제(1.1.43) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 iam:PassRole 와 datapipeline:CreatePipeline 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | iam:PassRole, datapipeline:CreatePipeline, datapipeline:PutPipelineDefinition 권한을 갖고 있는 경우, 임의의 AWS CLI 명령을 실행하거나 전달된 역할로 다른 리소스를 생성하도록 파이프라인을 생성한 후 이를 업데이트하여 권한 상승할 수 있다. 공격자는 파이프라인을 만들 때 전달된 역할과 관련된 권한에 액세스할 수 있으며, 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 iam:PassRole, datapipeline:CreatePipeline, datapipeline:PutPipelineDefinition 권한이 모두 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 iam:PassRole, datapipeline:CreatePipeline, datapipeline:PutPipelineDefinition 권한 중 하나라도 할당되어 있는 경우  ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 iam:PassRole, datapipeline:CreatePipeline, datapipeline:PutPipelineDefinition 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | lambda:UpdateFunctionCode 권한 해제(1.1.44) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 lambda:UpdateFunctionCode 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | lambda:UpdateFunctionCode 권한을 갖고 있는 경우, Lambda 함수의 코드를 업데이트할 수 있다. 이를 통해 해당 Lambda 서비스 역할과 연결된 권한에 대한 액세스 권한이 사용자에게 부여되어 권한 상승이 없는 사용자, 전체 관리자 계정 등에 접근할 수 있다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 lambda:UpdateFunctionCode 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 lambda:UpdateFunctionCode 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Policies)'을 클릭한다. 3. 해당 정책을 검색하여 클릭한다.  4. 정책 사용(Policy usage) 탭을 클릭한다. 5. 연결된 사용자 또는 그룹을 선택한 후 ,'분리(Detach)' 버튼을 클릭한다.  [AWS CLI] 1. 해당 정책에 연결된 모든 IAM 사용자, 그룹 및 역할을 나열한다.  # aws iam list-entities-for-policy --policy-arn <policy\_arn>   2. 모든 IAM 사용자들을 해당 정책에서 분리시킨다.  # aws iam detach-user-policy --user-name <iam\_user> --policy-arn <policy\_arn>   3. 모든 IAM 그룹들을 해당 정책에서 분리시킨다.  # aws iam detach-group-policy --group-name <iam\_group> --policy-arn <policy\_arn>   4. 모든 IAM 역할들을 해당 정책에서 분리시킨다. # aws iam detach-role-policy --role-name <iam\_role> --policy-arn <policy\_arn> | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 정책 목록을 가져온다.  # aws iam list-policies --only-attached --output text  2. 반환된 각 정책에 대해 lambda:UpdateFunctionCode 권한을 허용하는 정책이 있는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/IAM/latest/UserGuide/best-practices.html#grant-least-privilege https://aws.amazon.com/ko/premiumsupport/knowledge-center/iam-permission-boundaries/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 고객 관리형 정책에서 kms:ReEncryptFrom, kms:Decrypt 권한 해제(1.1.45) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자 또는 그룹에 적용된 고객 관리형 정책에서 kms:ReEncryptFrom, kms:Decrypt 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | IAM 고객 관리형 정책이 KMS(Key Management Service) 키에 대한 암호 해독 작업(kms:ReEncryptFrom, kms:Decrypt)을 허용하지 않는지 확인하는 것을 권장한다. AWS KMS를 사용할 경우, 고객 관리형 키(Customer-managed key)를 사용할 수 있는 사용자를 제어하고, 암호화된 데이터에 접근할 수 있다. 권한없는 사용자의 접근과 광범위한 권한 부여를 제한하기 위해 최소 권한의 원칙에 따라 적절한 정책만 암호 해독 권한을 갖는지 확인해야 한다. | | |
| 진단 기준 | ▶ 양호 : IAM 사용자 또는 그룹에 적용된 고객 관리형 정책에 kms:ReEncryptFrom, kms:Decrypt 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 사용자 또는 그룹에 적용된 고객 관리형 정책에 kms:ReEncryptFrom, kms:Decrypt 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 좌측 사이드 바에서 '정책(Users)'을 클릭한다.  3. 정책 필터(Filter policies)를 클릭하여 유형(Type)에서 '고객 관리형(Customer managed)'을 선택한다.  4. 각 고객 관리형 정책 좌측의 화살표 아이콘을 클릭한다.  5. 정책 편집(Edit policy) 버튼을 클릭한다.  6. KMS를 선택한다.  7. 작업(Actions)에서 '모든 KMS 작업(kms:\*)'의 체크박스를 해제한 후, 사용자에게 필요한 권한만을 할당한다. 또한, '쓰기(Write)' 작업 중 "kms:Decrypt"와 "kms:ReEncryptFrom"의 체크박스를 해제한다.  8. 정책 검토(Review policy) 버튼을 클릭한다.  9. 변경 사항 저장(Save changes) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '정책(Users)' 을 클릭한다.  3. 정책 필터(Filter policies)를 클릭하여 유형(Type)에서 '고객 관리형(Customer managed)' 을 선택한다.  4. 각 고객 관리형 정책 좌측의 화살표 아이콘을 클릭한다.  5. json 파일에 KMS 키에 대한 암호 해독 작업("kms:Decrypt" 와 "kms:ReEncryptFrom")을 허용하는 Actions이 "Allow" 로 설정되어 있는지 확인한다. 또는 "kms:\*" 권한이 설정되어 있는지 확인한다.   예시는 다음과 같다.  {  "Effect": "Allow",  "Action": [  "kms:Decrypt",  "kms:ReEncryptFrom"  ],  "Resource": "<optional>" }  ※ <optional>은 각 정책마다 다를 수 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 고객 관리형 정책 목록을 가져온다.  # aws iam list-policies --scope "Local" --only-attached  2. 반환된 정책에 대해 kms:Decrypt, kms:ReEncryptFrom 권한을 허용하는지 확인한다. # aws iam get-policy-version --policy-arn <policy\_arn> --version-id <version> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/access\_policies\_managed-vs-inline.html#customer-managed-policies https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-kms-11 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 인라인 정책에서 kms:ReEncryptFrom, kms:Decrypt 권한 해제(1.1.46) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 자격 증명(사용자, 그룹, 역할)에 할당된 인라인 정책에서 kms:ReEncryptFrom, kms:Decrypt 권한을 설정하지 않는 것을 권장한다. | | |
| 설명 | IAM 자격 증명(사용자, 그룹, 역할)에 포함된 인라인 정책이 KMS(Key Management Service) 키에 대한 암호 해독 작업(kms:ReEncryptFrom, kms:Decrypt)을 허용하지 않는지 확인하는 것을 권장한다. AWS KMS를 사용할 경우, 고객 관리형 키(Customer-managed key)를 사용할 수 있는 사용자를 제어하고, 암호화된 데이터에 접근할 수 있다. 권한 없는 사용자의 접근과 광범위한 권한 부여를 제한하기 위해 최소 권한의 원칙에 따라 적절한 IAM 자격증명만 암호 해독 권한을 갖는지 확인해야 한다. | | |
| 진단 기준 | ▶ 양호 : IAM 자격 증명(사용자, 그룹, 역할)에 할당된 인라인 정책에 kms:ReEncryptFrom, kms:Decrypt 권한이 할당되어 있지 않은 경우   ▶ 취약 : IAM 자격 증명(사용자, 그룹, 역할)에 할당된 인라인 정책에 kms:ReEncryptFrom, kms:Decrypt 권한이 할당되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 좌측 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 클릭한다.  4. 권한(Permissions) 탭을 선택한다.  5. 정책 유형이 '인라인 정책(Inline policy)'인 정책 좌측의 화살표 아이콘을 클릭한다.  6. 정책 편집(Edit policy) 버튼을 클릭한다.  7. 작업(Actions)에서 '모든 KMS 작업(kms:\*)'의 체크박스를 해제한 후, 사용자에게 필요한 권한만을 할당한다. 또한, '쓰기(Write)' 작업 중 "kms:Decrypt"와 "kms:ReEncryptFrom"의 체크박스를 해제한다.  8. 정책 검토(Review policy) 버튼을 클릭한다.  9. 변경 사항 저장(Save changes) 버튼을 클릭한다.  \* IAM 자격 증명(사용자, 그룹, 역할)의 모든 인라인 정책을 수정한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 사이드 바에서 '사용자(Users)'를 클릭한다.  3. 각 사용자를 클릭한다.  4. 권한(Permissions) 탭을 선택한다.  5. 정책 유형이 '인라인 정책(Inline policy)'인 정책 좌측의 화살표 아이콘을 클릭한다.  6. json 파일에 KMS 키에 대한 암호 해독 작업("kms:Decrypt" 와 "kms:ReEncryptFrom")을 허용하는 Actions이 "Allow" 로 설정되어 있는지 확인한다. 또는 "kms:\*" 권한이 설정되어 있는지 확인한다.   예시는 다음과 같다.  {  "Effect": "Allow",  "Action": [  "kms:Decrypt",  "kms:ReEncryptFrom"  ],  "Resource": "<optional>" }  ※ <optional>은 각 정책마다 다를 수 있다.  ※ IAM 자격 증명(사용자, 그룹, 역할)의 모든 인라인 정책을 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 IAM 사용자 목록을 출력한다.  # aws iam list-users --output text   2. 각 사용자에 할당된 인라인 정책을 출력한다.  # aws iam list-user-policies --user-name <iam user name>  3. 인라인 정책에 대해 kms:Decrypt, kms:ReEncryptFrom 권한을 허용하는지 확인한다. # aws iam get-user-policy --user-name <iam user name> --policy-name <inline policy name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/access\_policies\_managed-vs-inline.html#inline-policies https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-kms-2 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | IAM 사용자에 태그 지정(1.1.47) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | IAM | | |
| 소분류 | Permissions | | |
| 요약 | IAM 사용자에 대한 액세스 제어를 위해 태그를 지정하는 것을 권장한다. | | |
| 설명 | IAM 태그는 사용자에 추가할 수 있는 키-값 페어로 태그를 통해 사용자에 대한 액세스를 구성, 추적, 제어할 수 있다. 태그에 사용자의 이름, 이메일, 부서 등의 정보를 등록해 놓을 경우, 이를 기반으로 사용자가 수행할 수 있는 권한을 제어할 수 있다.  ※ 태그를 기반으로 액세스를 제어하기 위해서는 정책의 조건 요소에 태그를 등록해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 IAM 사용자에 태그(이름, 이메일, 부서 등)가 지정되어 있는 경우   ▶ 취약 : 각 IAM 사용자에 태그(이름, 이메일, 부서 등)가 지정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 좌측 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 해당 사용자를 클릭한다. 4. 태그(Tags) 탭을 선택한다.  5. 태그 편집(Edit tags) 버튼을 클릭한다.  6. 태그 키와 태그 값에 사용자의 이름, 이메일, 부서 등의 정보를 기입한다.  ex) Name : HongGilDong, Email : gildong@naver.com, Department : Marketing 7. 변경 내용을 저장합니다(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 IAM 사용자에 태그를 지정한다.  # aws iam tag-user --user-name <iam user name> --tags '{"Key": "<key>", "Value": "<value>"}' | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/iam/ 를 통해 IAM 대시보드에 접속한다. 2. 좌측 사이드 바에서 '사용자(Users)'를 클릭한다. 3. 각 사용자를 클릭한다. 4. 태그(Tags) 탭을 선택한다.  5. 태그 키와 태그 값에 사용자의 이름, 이메일, 부서 등 정보가 기입되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 IAM 사용자 목록을 출력한다.  # aws iam list-users  2. 각 IAM 사용자에 태그가 지정되어 있는지 확인한다.  # aws iam list-user-tags --user-name <iam user name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/access\_iam-tags.html#access\_iam-tags\_control-principals https://docs.aws.amazon.com/cli/latest/reference/iam/list-user-tags.html https://docs.aws.amazon.com/cli/latest/reference/iam/tag-user.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 삭제가 예약된 키 여부 검토(1.2.1) | 위험도 | Low |
| 대분류 | Identity&Crypto | | |
| 중분류 | KMS | | |
| 요약 | 고객 관리형 키 중 삭제 대기 중인 키가 실제 삭제하기로 검토 완료된 키인지 확인한다. | | |
| 설명 | AWS KMS(Key Management Service)에서 고객 관리형 키(Customer managed key)를 삭제할 경우 해당 키로 암호화된 민감 정보에 대해 더이상 복호화할 수 없게 되므로 키를 삭제할 때 이를 인지해야 한다. | | |
| 진단 기준 | ▶ 양호 : 삭제 대기 중인 키가 담당자에게 검토 완료된 키인 경우   ▶ 취약 : 삭제 대기 중인 키가 담당자에게 검토 완료되지 않은 키인 경우   ▶ Default Value : - | | |
| 조치 방안 | 삭제 검토가 안된 키의 경우, 담당자에게 보고 후 다음과 같이 조치를 한다. [AWS Console] 1. https://console.aws.amazon.com/kms/home 를 통해 AWS Key Management Service에 접속한다. 2. 사이드 바에서 '고객 관리형 키(Customer managed keys)'를 클릭한다. 3. 해당 고객 관리형 키를 클릭한다.  4. 우측 상단의 '키 작업(Key actiions)' 메뉴를 선택하여, "키 삭제 취소(Cancel key deletion)"를 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/kms/home 를 통해 AWS Key Management Service에 접속한다. 2. 사이드 바에서 '고객 관리형 키(Customer managed keys)'를 클릭한다. 3. 고객 관리형 키 중 '상태(Status)'가 "삭제 대기 중(Pending deletion)"인 키가 없는지 확인한다. 4. 삭제 대기 중인 키가 있을 경우, 해당 담당자에게 실제 삭제하기로 검토 완료된 키인지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/kms/latest/developerguide/enabling-keys.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 고객 관리형 키 자동 교체 설정(1.2.2) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | KMS | | |
| 요약 | 암호화 및 복호화 작업을 수행 시 사용되는 백업 키는 주기적으로 교체하는 것을 권장한다. | | |
| 설명 | AWS KMS(Key Management Service)를 통해 고객은 고객 관리형 키(Customer managed keys)의 키 ID와 연결된 KMS 내에 저장된 주요 데이터의 백업 키를 교체할 수 있다. 암호화 및 복호화와 같은 암호화 작업을 수행 시 사용되는 백업 키는 주기적으로 교체하는 것을 권장한다. 자동 키 교체는 암호화된 데이터의 복호화가 투명하게 이루어질 수 있도록 현재 모든 이전 백업 키를 보관한다. 고객 관리형 키 교체를 활성화하여 CMK가 손상되더라도 피해를 줄일 수 있도록 해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 고객 관리형 키에 "매년 이 CMK를 자동으로 교체합니다(Automatically rotate this CMK every year)" 옵션이 설정되어 있는 경우   ▶ 취약 : 각 고객 관리형 키에 "매년 이 CMK를 자동으로 교체합니다" 옵션이 설정되어 있지 않은 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/kms/home 를 통해 Key Management Service(KMS)에 접속한다. 2. 좌측 사이드 바에서 '고객 관리형 키(Customer managed keys)'를 클릭한다.  3. 해당 키를 클릭한다.  4. 키 교체(Key rotation) 탭을 클릭한다.  5. '매년 이 CMK를 자동으로 교체합니다(Automatically rotate this CMK every year)' 옵션을 설정한 후, 저장(Save)을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 키를 교체시킨다. # aws kms enable-key-rotation --key-id <kms\_key\_id> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/kms/home 를 통해 Key Management Service(KMS) 에 접속한다. 2. 좌측 사이드 바에서 '고객 관리형 키(Customer managed keys)'를 클릭한다.  3. 각 키를 클릭한다.  4. 키 교체(Key rotation) 탭을 클릭한다.  5. '매년 이 CMK를 자동으로 교체합니다(Automatically rotate this CMK every year)' 옵션이 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 키 목록을 출력한다.  # aws kms list-keys   2. 각 키에 대해 "KeyRotationEnabled" 속성값이 true 로 설정되어 있는지 확인한다. # aws kms get-key-rotation-status --key-id <kms\_key\_id> | | |
| Reference | 3.8 Ensure rotation for customer created CMKs is enabled (Automated) | | |
| Site | https://aws.amazon.com/ko/kms/pricing/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Secrets Manager 보안 암호 자동 교체 활성화(1.3.1) | 위험도 | Medium |
| 대분류 | Identity&Crypto | | |
| 중분류 | Secrets Manager | | |
| 요약 | 자격 증명을 안전하게 유지하기 위해 보안 암호가 자동 교체되도록 설정하는 것을 권장한다. | | |
| 설명 | AWS Secrets Manager의 보안 암호(secrets)의 자동 교체가 활성화되어 있는지 확인하는 것을 권장한다. 자동 교체를 활성화할 경우, 자격 증명을 정기적으로 변경하여 IT 리소스 및 데이터를 안전하게 유지할 수 있으며 오랜 기간 동안 변경되지 않은 자격 증명의 유출 위험을 줄일 수 있다. 또한, 교체 간격을 선택하여 지정된 일정마다 보안 암호를 교체할 수 있다.  ※ 애플리케이션에서 이 보안 암호를 사용하고 있지 않은 경우에 대해서만 자동 교체를 활성화하는 것을 권장한다. 애플리케이션에서 보안 암호를 사용하고 있고 AWS Secrets Manager를 사용하도록 애플리케이션을 업데이트 하지 않은 경우는 권장하지 않는다. | | |
| 진단 기준 | ▶ 양호 : 각 보안 암호의 자동 교체가 활성화되어 있는 경우   ▶ 취약 : 각 보안 암호의 자동 교체가 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/secretsmanager/ 를 통해 AWS Secrets Manager에 접속한다.  2. 좌측 사이드 바에서 '보안 암호(Secrets)'를 클릭한다.  3. 해당 보안 암호를 클릭한다.  4. 교체 편집(Edit rotation) 버튼을 클릭한다.  5. "자동 교체 활성화(Enable automatic rotation)"를 체크한 후, 교체 간격(rotation interval)을 알맞게 설정한다.  6. 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 보안 암호의 자동 교체를 활성화한다.  # aws secretsmanager rotate-secret --secret-id <secret id> --rotation-lambda-arn <lambda function arn> --rotation-rules AutomaticallyAfterDays=<value> ※ <value> 값은 1~365일 값이어야 하며, 30, 60, 90일 단위로 교체하는 것을 권장한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/secretsmanager/ 를 통해 AWS Secrets Manager에 접속한다.  2. 좌측 사이드 바에서 '보안 암호(Secrets)'를 클릭한다.  3. 각 보안 암호를 클릭한다.  4. 교체 구성(Rotation configuration) 항목에서 "교체 상태(Rotation status)"가 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 보안 암호의 "RotationEnabled" 항목이 "true"로 설정되어 있는지 확인한다.  # aws secretsmanager list-secrets | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/secretsmanager/latest/userguide/terms-concepts.html?icmpid=docs\_asm\_console#term\_rotation https://docs.aws.amazon.com/secretsmanager/latest/userguide/rotating-secrets.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/secretsmanager-rotation-enabled-check.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-secretsmanager-1 | | |

**Compute**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EC2 인스턴스 세부 모니터링 활성화(2.1.1) | 위험도 | Low |
| 대분류 | Compute | | |
| 중분류 | EC2 | | |
| 요약 | EC2 인스턴스의 세부 모니터링을 활성화하는 것을 권장한다. | | |
| 설명 | EC2의 세부 모니터링을 활성화할 경우, Amazon EC2 콘솔에 인스턴스에 대한 1분 모니터링 그래프를 표시해주며 유사한 인스턴스 그룹 간에 집계된 데이터를 제공한다. 리소스의 안정성, 가용성 및 성능을 유지하기 위헤 세부 모니터링을 설정하는 것을 권장한다.  ※ 인스턴스에 대한 세부 모니터링을 활성화할 경우, IAM 사용자에게 MonitorInstances API 작업을 사용할 권한이 필요하며 추가적인 요금이 부과될 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 인스턴스의 세부 모니터링(Detailed monitoring)이 활성화되어 있는 경우    ▶ 취약 : 각 EC2 인스턴스의 세부 모니터링이 비활성화되어 있는 경우    ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 클릭한다.  4. 모니터링(Monitoring) 탭을 선택한다.  5. 세부 모니터링 관리(Manage detailed monitoring) 버튼을 클릭한다.  6. 세부 모니터링(Detailed monitoring)의 "활성화(Enable)"를 체크한다.  7. 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EC2 인스턴스의 세부 모니터링을 활성화한다.  # aws ec2 monitor-instances --instance-ids <instance id> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 세부 정보(Details) 탭을 선택한다.  5. 모니터링(Monitoring) 항목이 "세부 정보(detailed)"로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 "Monitoring" 항목의 "State"가 "enabled"로 설정되어 있는지 확인한다.  # aws ec2 describe-instances --instance-ids <instance id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-cloudwatch-new.html https://aws.amazon.com/ko/cloudwatch/pricing/ https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/ec2-instance-detailed-monitoring-enabled.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EC2 종료 방지 활성화(2.1.2) | 위험도 | Medium |
| 대분류 | Compute | | |
| 중분류 | EC2 | | |
| 요약 | EC2 인스턴스의 종료 방지 기능을 활성화하여 사용자의 실수로 데이터가 삭제되는 것을 보호하는 것을 권장한다. | | |
| 설명 | EC2 인스턴스의 종료(terminate)는 더이상 필요하지 않은 인스턴스를 삭제하는 것을 의미하며 다시 연결하거나 재시작할 수 없다. 인스턴스를 종료할 경우, 해당 인스턴스와 관련된 모든 인스턴스 스토어 볼륨의 데이터가 삭제되며, EBS 루트 디바이스 볼륨은 자동으로 삭제된다. 하지만 시작 시 연결하는 추가 EBS 볼륨 또는 기존 인스턴스에 연결하는 EBS 볼륨은 인스턴스가 종료된 후에도 유지된다. 인스턴스가 삭제될 때 볼륨의 삭제 여부는 "DeleteOnTermination" 속성에 따라 결정된다. 종료 방지 기능을 활성화할 경우, 인스턴스가 갑작스럽게 종료되는 것을 방지할 수 있으며 해당 기능이 비활성화될 때까지 인스턴스를 종료할 수 없다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 인스턴스의 "종료 방지(Termination protection)" 항목이 활성화되어 있는 경우   ▶ 취약 : 각 EC2 인스턴스의 "종료 방지" 항목이 비활성화되어 있는 경우  ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 체크한다.  4. 작업(Actions) 탭의 '인스턴스 설정(Instance settings)'의 '종료 방지 기능 변경(Change termination protection)'을 클릭한다.  5. 종료 방지(Termination protection) 항목의 "활성화(Enable)"를 체크한다.  6. 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EC2 인스턴스의 종료 방지를 활성화한다.  # aws ec2 modify-instance-attribute --instance-id <instance id> --disable-api-termination | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 체크한다.  4. 작업(Actions) 탭의 '인스턴스 설정(Instance settings)'의 '종료 방지 기능 변경(Change termination protection)'을 클릭한다.  5. 종료 방지(Termination protection) 항목의 "활성화(Enable)"가 체크되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EC2 인스턴스의 종료 방지가 활성화되어 있는지 확인한다.  # aws ec2 describe-instance-attribute --instance-id <instance id> --attribute disableApiTermination | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/terminating-instances.html#Using\_ChangingDisableAPITermination https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/terminating-instances.html#preserving-volumes-on-termination | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EC2 리소스 태그 지정(2.1.3) | 위험도 | Low |
| 대분류 | Compute | | |
| 중분류 | EC2 | | |
| 요약 | 각 EC2 리소스에 태그를 지정하는 것을 권장한다. | | |
| 설명 | 태그는 AWS 리소스에 할당하는 키-값 쌍이며, 태그를 통해 용도, 소유자 또는 환경을 기준으로 AWS 리소스를 다양한 방식으로 분류할 수 있다. 태그를 설정할 경우, 인스턴스, 이미지 및 기타 Amazon EC2 리소스를 쉽게 관리 및 식별할 수 있다. 또한, 리소스 태그를 사용하여 속성 기반 제어(ABAC)를 구현하거나 태그를 기반으로 작업을 허용하는 IAM 정책을 생성하여 액세스를 제어할 수 있다. AWS 청구서를 태그 별로 구분하여 분석할 수도 있다. ※ 사용자 지정 태그 식별자는 대/소문자를 구분한다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 리소스에 태그가 지정되어 있는 경우   ▶ 취약 : 각 EC2 리소스에 태그가 지정되어 있지 않은 경우  ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 클릭한다.  4. 태그(Tags) 탭을 클릭한다.  5. 태그 관리(Manage tags) 버튼을 클릭한다.  6. 태그 추가(Add tag) 버튼을 클릭하여, 키(Key)와 값(Value)을 입력한다.  7. 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EC2 리소스에 태그를 지정한다.  # aws ec2 create-tags --resources <resource id> --tags Key=<tag key>,Value=<tag value> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 태그(Tags) 탭을 클릭한다.  5. 리소스에 지정된 태그가 존재하는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EC2 리소스에 태그가 지정되어 있는지 확인한다.  # aws ec2 describe-tags --filters "Name=resource-id,Values=<resource id>" | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Using\_Tags.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EC2 인스턴스에 키 페어 액세스 설정(2.1.4) | 위험도 | Medium |
| 대분류 | Compute | | |
| 중분류 | EC2 | | |
| 요약 | EC2 인스턴스에 키 페어를 연결하여 액세스하는 것을 권장한다. | | |
| 설명 | 키 페어는 EC2 인스턴스에 연결할 때 인증에 필요한 보안 자격 증명으로 AWS에 저장하는 퍼블릭 키와 사용자가 저장하는 프라이빗 키로 구성되어 있다. 암호 대신 프라이빗 키를 사용하여 SSH를 통해 인스턴스에 안전하게 접속할 수 있다. 프라이빗 키를 소유한 사람은 누구나 인스턴스에 연결할 수 있으므로 사용자만이 액세스 할 수 있는 안전한 위치에 저장해야 하며, 잃어버릴 경우 복구할 수 없다. EC2에서 사용되는 키는 2048비트 SSH-2 RSA 키이며, 리전당 최대 5,000개의 키 페어를 가질 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 인스턴스가 키 페어와 연결되어 있고, 키 페어로 인스턴스에 접근하는 경우  ▶ 취약 : 각 EC2 인스턴스가 키 페어와 연결되어 있지 않거나 키 페어가 아닌 일반 패스워드로 인스턴스에 접근하는 경우  ▶ Default Value : - | | |
| 조치 방안 | ※ 키 페어가 연결되어 있지 않은 기존 인스턴스를 사용하고자 하는 경우 [AWS Console]  1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 클릭한다.  4. 인스턴스 상태(Instance state) 버튼을 클릭하여 '인스턴스 중지(Stop instance)'를 선택한다.  5. 인스턴스가 완전히 중지되면, 좌측 메뉴바의 '인스턴스(Instances)'를 다시 클릭한다.  6. 해당 인스턴스를 체크한 후, '작업(Actions)' 버튼을 클릭하여 '이미지 및 템플릿(Image and templates)'의 '이미지 생성(Create image)'을 클릭한다.  7. 생성할 AMI의 이름, 정보 등을 설정한 후 '이미지 생성(Create image)' 버튼을 클릭한다.  8. 우측 메뉴바의 '이미지(Images)'의 'AMI(AMIs)'를 클릭한다.  9. 생성한 AMI 이미지를 체크한 후, '작업(Actions)' 버튼을 클릭하여 '시작하기(Launch)'를 선택한다.  10. 인스턴스 유형, 인스턴스 세부 정보, 스토리지, 보안 그룹 등을 설정한 후, 시작하기(Launch) 버튼을 클릭한다.  11. '기존 키 페어 또는 새 키 페어 생성(Select an existing key pair or create a new key pair)' 창에서 기존 키 페어를 선택하거나 새 키 페어를 생성하여 다운로드한다.  ※ 새 키 페어를 다운로드 받은 경우 안전한 곳에 보관한다.  12. 인스턴스 시작(Launch instances) 버튼을 클릭한다.   ※ 새 인스턴스를 생성하는 경우 [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다. 3. 인스턴스 시작(Launch instances) 버튼을 클릭한다.  4. AMI(Amazon Machine Image), 인스턴스 유형, 인스턴스 세부 정보, 스토리지, 보안 그룹 등을 설정한 후, 시작하기(Launch) 버튼을 클릭한다.  5. '기존 키 페어 또는 새 키 페어 생성(Select an existing key pair or create a new key pair)' 창에서 기존 키 페어를 선택하거나 새 키 페어를 생성하여 다운로드한다.  ※ 새 키 페어를 다운로드 받은 경우 안전한 곳에 보관한다.  6. 인스턴스 시작(Launch instances) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 세부 정보(Details) 탭을 클릭한다.  5. 인스턴스 세부 정보(Instance details)에서 "키 페어 이름(Key pair name)"이 지정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 "KeyName" 항목이 설정되어 있는지 확인한다.  # aws ec2 describe-instances --instance-ids <instance id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EC2 인스턴스 메타데이터 버전 설정(2.1.6) | 위험도 | Low |
| 대분류 | Compute | | |
| 중분류 | EC2 | | |
| 요약 | EC2의 인스턴스 메타데이터 버전을 IMDSv2로 설정하는 것을 권장한다. | | |
| 설명 | 인스턴스 메타데이터는 실행 중인 인스턴스를 구성 또는 관리하는데 사용될 수 있는 인스턴스 관련 데이터이다. IMDSv2(Instance Metadata Service Version 2)는 인스턴스에서 인스턴스 메타데이터에 접근할 수 있는 방법 중 세션 지향 방법이며, 세션 지향 요청을 사용하여 세션 기간을 정의하는 세션 토큰을 생성한다.   IMDSv2는 아래와 같은 취약성에 대한 보호 기능을 가지고 있다.  - 웹사이트 애플리케이션 방화벽 열기 - 역방향 프록시 열기 - SSRF(Server-Side Request Forgery) 취약점 - 3계층 방화벽 및 NAT(Network Address Translation) 열기  ※ 인스턴스에 직접 액세스할 수 있는 모든 사람과 인스턴스에서 실행 중인 모든 소프트웨어는 메타데이터를 볼 수 있으므로 암호 또는 암호화 키와 같은 민감 데이터를 사용자 데이터로 저장해서는 안된다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 인스턴스의 메타데이터 버전이 "IMDSv2"으로 설정되어 있는 경우  ▶ 취약 : 각 EC2 인스턴스의 메타데이터 버전이 "IMDSv1"으로 설정되어 있는 경우  ▶ Default Value : IMDSv1 | | |
| 조치 방안 | ※ 현재 인스턴스 메타데이터 옵션을 수정하는 것은 AWS Console에서는 지원하지 않으며, AWS SDK 또는 CLI 로만 가능하다.   [AWS CLI] 1. 다음 명령어를 실행하여 인스턴스 메타데이터 옵션을 수정한다.  # aws ec2 modify-instance-metadata-options --instance-id <instance id> --http-tokens required --http-endpoint enabled | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 각 인스턴스의 "MetadataOptions" 항목이 아래와 같이 설정되어 있는지 확인한다.  # aws ec2 describe-instances --instance-ids <instance id>  예시는 다음과 같다.  "MetadataOptions": {  "HttpTokens": "required",  "HttpPutResponseHopLimit": 1,  "HttpEndpoint": "enabled" } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-metadata.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-ec2-8 https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/ec2-imdsv2-check.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EBS 최적화 설정(2.1.8) | 위험도 | Low |
| 대분류 | Compute | | |
| 중분류 | EC2 | | |
| 요약 | EC2 인스턴스에 EBS 최적화 설정하는 것을 권장한다. | | |
| 설명 | Amazon EBS 최적화 인스턴스는 최적화된 구성 스택을 사용하며, EBS I/O를 위한 추가 전용 용량을 제공한다. 이를 통해 인스턴스에서 EBS I/O와 다른 트래픽간의 경합이 최소화되어 EBS 볼륨의 성능을 극대화할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 인스턴스에 "EBS 최적화(EBS optimization)"가 활성화되어 있는 경우   ▶ 취약 : 각 EC2 인스턴스에 "EBS 최적화"가 비활성화되어 있는 경우  ▶ Default Value : 인스턴스 유형에 따라 상이함 | | |
| 조치 방안 | ※ 기존 인스턴스에 대해 EBS 최적화를 활성화하고자 하는 경우, 실행 중인 인스턴스를 먼저 중지해야 한다.  ※ 인스턴스를 중지할 경우 인스턴스 스토어 볼륨의 데이터가 삭제된다. 인스턴스 스토어 볼륨의 데이터를 유지하기 위해서는 영구 스토리지에 백업해야 한다.   [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 체크한다.  4. 상단의 '작업(Actions)' 버튼을 클릭하여 '인스턴스 설정(Instance settings)'의 '인스턴스 유형 변경(Change instance type)'을 선택한다.  5. "EBS 최적화(EBS-optimized)" 항목의 체크박스를 체크한다.  6. 적용(Apply) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EBS 최적화를 활성화한다.  # aws ec2 modify-instance-attribute --instance-id <instance id> --ebs-optimized | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 스토리지(Storage) 탭을 선택한다.  5. 루트 디바이스 세부 정보(Root device name)에서 "EBS 최적화(EBS optimization)" 항목이 "활성(enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 "EbsOptimized" 항목이 "true"로 설정되어 있는지 확인한다.  # aws ec2 describe-instance-attribute --instance-id <instance id> --attribute ebsOptimized | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-optimized.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/ebs-optimized-instance.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Lambda 함수 인터넷 노출 제한(2.2.1) | 위험도 | Low |
| 대분류 | Compute | | |
| 중분류 | Lambda | | |
| 요약 | Lambda 함수가 외부에 노출될 경우 데이터 유출로 이어질 수 있다. | | |
| 설명 | 퍼블릭 액세스를 허용하거나 모든 사용자 또는 0.0.0.0/0 에서 접근 가능하도록 설정할 경우 어떠한 접근 제어 없이 인터넷에 노출되는 것과 같으며, 이는 데이터 유출로 연결될 수 있다. 적절한 권한을 부여하여 신뢰할 수 있는 엔티티에 의해서만 Lambda 함수를 호출하도록 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 Lambda 함수가 인터넷에서 접근 가능하지 않도록 설정되어 있는 경우   ▶ 취약 : 각 Lambda 함수가 인터넷에서 접근 가능하도록 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 보안 그룹 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 보안 그룹(Security Groups)에서 Lambda에 연결된 보안 그룹을 클릭한다.  6. 해당 보안 그룹으로 이동한다.  7. 인바운드 규칙(Inbound rules) 탭을 클릭하여, Source가 0.0.0.0/0 인 규칙을 삭제한다.  8. 아웃바운드 규칙(Outbound rules) 탭을 클릭하여, Destination이 0.0.0.0/0 인 규칙을 삭제한다.   ※ 라우팅 테이블 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 서브넷(Subnets)에서 Lambda에 연결된 서브넷을 클릭한다.  6. 해당 서브넷 ID를 클릭한다. 7. 라우팅 테이블(Route table) 탭을 선택한다.  8. "라우팅 테이블 연결 편집(Edit route table association)" 버튼을 클릭하여, 라우팅 테이블을 수정한다.   ※ 네트워크 ACL 확인 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 서브넷(Subnets)에서 Lambda에 연결된 서브넷을 클릭한다.  6. 해당 서브넷 ID를 클릭한다. 7. 네트워크 ACL(Network ACL) 탭을 선택한다.  8. 인바운드 규칙과 아웃바운드 규칙에서 해당 규칙을 삭제한다.   - 소스(Source) : 0.0.0.0/0  - 허용/거부(Allow/Deny) : Allow | | |
| 감사 방안 | [AWS Console] ※ 보안 그룹 확인 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 보안 그룹(Security Groups)에서 Lambda에 연결된 보안 그룹을 클릭한다.  6. 해당 보안 그룹으로 이동한다.  7. 인바운드 규칙(Inbound rules) 탭을 클릭하여, Source가 0.0.0.0/0 인 규칙이 존재하는지 확인한다.  8. 아웃바운드 규칙(Outbound rules) 탭을 클릭하여, Destination이 0.0.0.0/0 인 규칙이 존재하는지 확인한다.   ※ 라우팅 테이블 확인 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 서브넷(Subnets)에서 Lambda에 연결된 서브넷을 클릭한다.  6. 해당 서브넷 ID를 클릭한다. 7. 라우팅 테이블(Route table) 탭을 선택한다.  8. 아래와 같은 연결이 존재하는지 확인한다.   - Destination : 0.0.0.0/0  - Target : Internet Gateway 또는   - Destination : 0.0.0.0/0  - Target : NAT Gateway ※ Lambda에 연결된 모든 서브넷의 라우팅 테이블을 확인한다.   ※ 네트워크 ACL 확인 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 서브넷(Subnets)에서 Lambda에 연결된 서브넷을 클릭한다.  6. 해당 서브넷 ID를 클릭한다. 7. 네트워크 ACL(Network ACL) 탭을 선택한다.  8. 인바운드 규칙과 아웃바운드 규칙에 아래와 같은 규칙이 존재하는지 확인한다.   - 소스(Source) : 0.0.0.0/0  - 허용/거부(Allow/Deny) : Allow ※ Lambda에 연결된 모든 서브넷의 네트워크 ACL을 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://aws.amazon.com/ko/blogs/architecture/best-practices-for-developing-on-aws-lambda/ https://aws.amazon.com/ko/premiumsupport/knowledge-center/internet-access-lambda-function/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Lamdba 함수 VPC 액세스 설정(2.2.3) | 위험도 | Medium |
| 대분류 | Compute | | |
| 중분류 | Lambda | | |
| 요약 | Lambda 함수가 VPC에 액세스하도록 설정하는 것을 권장한다. | | |
| 설명 | Lambda는 여러 가용 영역에서 함수를 실행하여 단일 영역에서 서비스 중단이 발생한 경우 이벤트를 처리할 수 있도록 한다. Lambda를 VPC에 연결할 경우, 여러 가용 영역에 서브넷을 지정하여 고가용성을 보장할 수 있다. 또한, VPC를 통해 데이터베이스, 캐시 인스턴스, 내부 서비스 등과 같은 리소스에 대해 프라이빗 네트워크를 생성한 후 Lambda를 연결하여 프라이빗 리소스에 액세스할 수 있다.  ※ 프라이빗 리소스에 연결하려면 Lambda 함수를 프라이빗 서브넷에 액세스해야 한다. 함수에 인터넷 액세스 권한이 필요할 경우 NAT를 사용해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 Lambda 함수가 VPC에 액세스하도록 설정되어 있는 경우   ▶ 취약 : 각 Lambda 함수가 VPC에 액세스하도록 설정되어 있지 않은 경우   ▶ Default Value : Not set | | |
| 조치 방안 | ※ Lambda 함수를 VPC에 연결할 경우, 함수의 실행 역할에 ec2:CreateNetworkInterface, ec2:DescribeNetworkInterfaces, ec2:DeleteNetworkInterface 와 같은 권한이 필요하다. 해당 권한은 AWS 관리형 정책인 AWSLambdaVPCAccessExecutionRole에 포함되어 있다. 또한, IAM 사용자는 ec2:DescribeSecurityGroups, ec2:DescribeSubnets, ec2:DescribeVpcs 와 같은 권한이 필요하다.   [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. '편집(Edit)' 버튼을 클릭한다.  6. 연결할 VPC, 서브넷, 보안 그룹을 설정한 후, 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Lambda 함수에 VPC에 연결한다.  # aws lambda update-function-configuration --function-name <function name> --vpc-config SubnetIds=string,string, SecurityGroupIds=string,string ※ 서브넷의 경우, 2개 이상의 서브넷을 지정해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. VPC 탭을 클릭한다.  5. 연결된 VPC가 존재하는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Lambda 함수가 VPC에 연결되어 있는지 확인한다. "VpcConfig" 항목이 존재하지 않으면 VPC에 연결되어 있지 않음을 의미한다.   예시는 다음과 같다.  "VpcConfig": {  "SubnetIds": [  "<subnet id>",  "<subnet id>"  ],  "SecurityGroupIds": [  "<sg id>"  ],  "VpcId": "<vpc id>" } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/lambda/latest/dg/configuration-vpc.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/lambda-inside-vpc.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Lambda 함수에 배달 못한 편지 대기열 구성(2.2.4) | 위험도 | Medium |
| 대분류 | Compute | | |
| 중분류 | Lambda | | |
| 요약 | Lambda 함수에 배달 못한 편지 대기열 서비스를 구성하는 것을 권장한다. | | |
| 설명 | 비동기식 호출의 경우, 재시도가 모두 실패했을 때 배달 못한 편지 대기열(Dead-letter queue)로 요청을 전송하도록 Lambda를 구성할 수 있다. 배달 못한 편지 대기열은 문제 해결이나 재처리에 대한 이벤트를 수신하는 Amazon SNS 주제 또는 Amazon SQS 대기열이다. 애플리케이션의 내부 오류 또는 코드 삭제 등 여러가지 문제로 인해 처리될 수 없는 메시지를 따로 저장하여, 애플리케이션 및 메시징 시스템을 디버깅하거나 소프트웨어 또는 생산자, 사용자의 하드웨어 등 어느 지점에서 문제가 발생했는지 파악할 수 있다. SNS 주제에서 이벤트를 이메일 주소, Lambda 함수 또는 HTTP 엔드포인트로 전송하도록 구성할 수 있으며, SQS 대기열에서 실패한 이벤트를 보관할 수 있다. 배달 못한 편지 대기열 메시지에는 호출 요청의 ID, HTTP 상태 코드(에러 코드), 에러 메시지의 첫 1KB가 포함된다.  ※ SNS 주제 또는 SQS 대기열로 이벤트를 전송하기 위해서는 함수에 sqs:SendMessage, sns:Publish와 같은 추가적인 실행 역할이 필요하다. | | |
| 진단 기준 | ▶ 양호 : 각 Lambda 함수에 "배달 못한 편지 대기열 서비스(Dead-letter queue service)"가 Amazon SNS 또는 Amazon SQS로 설정되어 있는 경우   ▶ 취약 : 각 Lambda 함수에 "배달 못한 편지 대기열 서비스"가 설정되어 있지 않은 경우   ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 해당 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. 비동기식 호출(Asynchronous invocation) 탭을 클릭한다.  5. 편집(Edit) 버튼을 클릭한다.  6. "배달 못한 편지 대기열 서비스(Dead-letter queue service)" 항목에서 "Amazon SNS" 또는 "Amazon SQS"로 선택한 후, 주제 또는 대기열을 설정한다.  7. 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 Lambda 함수에 배달 못한 편지 대기열 서비스를 설정한다.  # aws lambda update-function-configuration --function-name <function name> --dead-letter-config TargetArn=<sns arn or sqs arn> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. 비동기식 호출(Asynchronous invocation) 탭을 클릭한다.  5. "배달 못한 편지 대기열 서비스(Dead-letter queue service)" 항목이 "Amazon SNS" 또는 "Amazon SQS"로 설정되어 있는지 확인한다.   [AWS CLI]  1. 다음 명령어를 실행하여 각 Lambda 함수에 배달 못한 편지 대기열 서비스가 설정되어 있는지 확인한다. "DeadLetterConfig" 항목이 존재하지 않은 경우, 배달 못한 편지 대기열 서비스가 설정되지 않았음을 의미한다.  # aws lambda get-function-configuration --function-name <function name>  예시는 다음과 같다.  "DeadLetterConfig": {  "TargetArn": "<sns arn or sqs arn>" } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/lambda/latest/dg/invocation-async.html https://docs.aws.amazon.com/lambda/latest/dg/invocation-async.html#dlq https://docs.aws.amazon.com/config/latest/developerguide/lambda-dlq-check.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/lambda/latest/dg/security-resilience.html https://docs.aws.amazon.com/ko\_kr/lambda/latest/dg/lambda-intro-execution-role.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Lambda 함수의 리소스 기반 정책 설정(2.2.5) | 위험도 | High |
| 대분류 | Compute | | |
| 중분류 | Lambda | | |
| 요약 | Lambda의 리소스 기반 정책을 통해 퍼블릭 액세스를 차단하는 것을 권장한다. | | |
| 설명 | AWS Lambda는 Lambda 함수 및 계층에 대한 리소스 기반 권한 정책을 지원한다. 이를 통해 리소스별로 다른 AWS 계정에 사용 권한을 부여하거나 AWS 서비스가 사용자를 대신하여 함수를 호출할 수 있다. Lambda 콘솔에서 생성하는 기본 실행 역할에는 Amazon CloudWatch Logs에 로그를 저장하는 권한만 포함되어 있다. AWS SDK를 사용하여 AWS 서비스를 호출할 경우, 해당 서비스의 API 작업에 대한 액세스 권한을 부여하는 정책을 추가해야하며 X-Ray 추적과 같은 일부 기능에는 추가 권한이 필요하다. Lambda의 리소스 기반 정책을 통해 함수에 대한 액세스 권한이 필요한 사용자 및 서비스에게만 할당하여 퍼블릭 액세스를 차단하는 것을 권장한다. 퍼블릭 액세스가 허용될 경우 리소스 가용성 저하로 이어질 수 있으므로 Lambda 함수에 대한 액세스를 관리해야 한다.  ※ 리소스 기반 정책은 단일 기능, 버전, 별칭 또는 계층 버전에 적용되며, 여러 리소스에 대한 액세스 권한을 갖거나 리소스 기반 정책이 지원하지 않는 API 작업을 사용하려는 신뢰할 수 있는 계정의 경우 교차 계정 역할을 사용해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 Lambda 함수의 리소스 기반 정책에 액세스가 필요한 사용자 및 서비스에게만 권한이 적절하게 할당되어 있는 경우   ▶ 취약 : 각 Lambda 함수의 리소스 기반 정책에 액세스가 필요하지 않는 사용자 및 서비스에게 권한이 할당되어 있는 경우    ▶ Default Value : Not set | | |
| 조치 방안 | ※ 권한 추가 [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 해당 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. 권한(Permissions) 탭을 클릭한다.  5. 리소스 기반 정책(Resource-based policy)에서 "권한 추가(Add permissions)" 버튼을 클릭한다.  6. 권한을 부여할 AWS 계정 또는 AWS 서비스를 선택한 후, 보안 주체의 ARN을 작성한다.  7. 허용할 작업과 정책 ID를 작성한 후, 저장(Save) 버튼을 클릭한다.   ※ 권한 삭제  [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 해당 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. 권한(Permissions) 탭을 클릭한다.  5. 리소스 기반 정책(Resource-based policy) 에서 필요하지 않은 정책 문을 선택한 후, "삭제(Delete)" 버튼을 클릭한다.  6. 제거(Remove) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 리소스 기반 정책의 정책 문을 삭제한다.  # aws lambda remove-permission --function-name <function name> --statement-id <id> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/lambda/ 를 통해 AWS Lambda에 접속한다.  2. 각 Lambda 함수를 클릭한다.  3. 구성(Configuration) 탭을 선택한다. 4. 권한(Permissions) 탭을 클릭한다.  5. 리소스 기반 정책(Resource-based policy)에 Lambda 함수에 대한 액세스가 필요한 사용자 및 서비스에게만 권한이 적절하게 할당되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 Lambda 함수에 대한 액세스가 필요한 사용자 및 서비스에게만 권한이 적절하게 할당되어 있는지 확인한다.  # aws lambda get-policy --function-name <function name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/lambda/latest/dg/access-control-resource-based.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/lambda-function-public-access-prohibited.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-lambda-1 | | |

**Storage**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 기본 암호화 설정(3.1.1) | 위험도 | High |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | Amazon S3 버킷에 데이터 보관 시 AES-256 이상으로 암호화하는 것을 권장한다. | | |
| 설명 | S3 버킷의 기본 암호화를 활성화할 경우, 데이터가 의도치 않게 노출될 가능성이 낮아지며 유출되었더라도 복호화되지 않는다면 영향을 무효화할 수 있다. SSE-KMS를 사용한 기본 버킷 암호화를 사용하는 AWS S3 버킷은 Amazon S3 서버 액세스 로깅의 대상 버킷으로 사용할 수 없다. 서버 액세스 로그 대상 버킷은 SSE-S3 기본 암호화만 지원한다.  ※ S3 버킷 암호화는 버킷에 저장된 새 객체에만 적용되며 암호화 설정 이전에 저장된 객체에는 적용되지 않는다. | | |
| 진단 기준 | ▶ 양호 : 버킷의 기본 암호화(Default encryption)가 SSE-S3 또는 SSE-KMS로 설정되어 있는 경우   ▶ 취약 : 버킷의 기본 암호화가 비활성화되어있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다. 4. "기본 암호화(Default encryption)"의 '편집(Edit)' 버튼을 클릭한다.  5. 서버 측 암호화(Server-side encryption)에서 "활성화(Enable)"를 선택한다.  6. SSE-S3 또는 SSE-KMS로 선택한 후 '변경 사항 저장(Save changes)'을 클릭한다.  [AWS CLI] 1. 다음 명령어를 통해 SSE-S3 또는 SSE-KMS로 설정한다. # aws s3api put-bucket-encryption --bucket <bucket name> --server-side-encryption-configuration '{"Rules": [{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm": "AES256"}}]}'  또는   # aws s3api put-bucket-encryption --bucket <bucket name> --server-side-encryption-configuration '{"Rules": [{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm": "aws:kms","KMSMasterKeyID": "aws/s3"}}]}' | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다. 4. "기본 암호화(Default encryption)" 가 SSE-S3 또는 SSE-KMS로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷 목록을 출력한다.  # aws s3 ls  2. 각 버킷 마다 "SSEAlgorithm": "AES256" 또는 "SSEAlgorithm": "aws:kms"로 설정되어 있는지 확인한다. # aws s3api get-bucket-encryption --bucket <bucket name> | | |
| Reference | 2.1.1 Ensure all S3 buckets employ encryption-at-rest (Manual) | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/user-guide/default-bucket-encryption.html https://docs.aws.amazon.com/AmazonS3/latest/dev/Welcome.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-server-side-encryption-enabled.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-s3-4 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 정책 HTTP 요청 거부 설정(3.1.2) | 위험도 | High |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | Amazon S3 버킷 수준에서 버킷 정책으로 사용 권한을 구성하여 HTTPS를 통해서만 객체에 접근할 수 있도록 설정한다. | | |
| 설명 | Amazon S3는 기본적으로 HTTP와 HTTPS 요청을 모두 허용한다. 전송 중 암호화를 위해 HTTPS 요청만을 허용하고, HTTP 요청에 대해서는 명시적으로 거부하는 것을 권장한다. HTTP 요청을 명시적으로 거부하지 않고 HTTPS 요청만 허용하도록 설정할 경우, HTTPS를 통해 전송되지만 HTTP 또한 계속 사용할 수 있다. 따라서 HTTP 요청으로 명시적으로 거부하도록 버킷 정책을 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 버킷 정책 구문에 "aws:SecureTransport: false"이고 "Effect: deny"로 설정되어 있는 경우   ▶ 취약 : 버킷 정책 구문에 "aws:SecureTransport: false"이고 "Effect: deny"로 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 권한(Permissions) 탭을 클릭한다.  4. "버킷 정책(Bucket policy)"의 '편집(Edit)'을 클릭한다. 5. '정책 생성기(Policy generator)' 버튼을 클릭한다. 6. Select Type of Policy 에서 "S3 Bucket Policy"를 선택한다. 7. Add Statement(s) 에서 다음과 같이 입력 후, Add Statement 버튼을 클릭한다.  - Effect = Deny  - Principal = \*  - AWS Service = Amazon S3   - Actions = GetObject   - Amazon Resource Name = <bucket\_arn>  8. Generate Policy 를 클릭한다. 생성된 값의 예는 다음과 같다. {  "Version": "2012-10-17",  "Id": "Policy1617601575524",  "Statement": [  {  "Sid": "Stmt1617600009898",  "Effect": "Deny",  "Principal": "\*",  "Action": "s3:GetObject",  "Resource": "arn:aws:s3:::sample\_bucket/\*"  }  ] }  9. 해당 구문을 복사하여 버킷 정책에 추가한 후, '변경 사항 저장(Save changes)'을 클릭한다.  [AWS CLI] 1. 버킷 정책을 json 파일로 추출한다. # aws s3api get-bucket-policy --bucket <bucket\_name> --query Policy --output text > policy.json  2. policy.json 파일의 Statement에 다음과 같은 구문을 추가한다. # vi policy.json  {  "Sid": <optional>",  "Effect": "Deny",  "Principal": "\*",  "Action": "s3:GetObject",  "Resource": "arn:aws:s3:::<bucket\_name>/\*",   "Condition": {  "Bool": {  "aws:SecureTransport": "false"  }   } }  3. 수정된 정책을 S3 버킷에 적용한다. # aws s3api put-bucket-policy --bucket <bucket\_name> --policy file://policy.json | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 권한(Permissions) 탭을 클릭한다.  3. "버킷 정책(Bucket policy)" 의 Statement에 다음과 같은 구문이 포함되어 있는지 확인한다.  {  "Sid": <optional>,   "Effect": "Deny",   "Principal": "",  "Action": "s3:GetObject",  "Resource": "arn:aws:s3:::<bucket\_name>/",   "Condition": {  "Bool": {  "aws:SecureTransport": "false"  }  } } ※ <optional>과 <bucket\_name>은 계정에 따라 다르다.  [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷 목록을 출력한다.  # aws s3 ls  2. 각 버킷마다 'aws:SecureTransport:false' 이고 'Effect:Deny' 인지 확인한다. # aws s3api get-bucket-policy --bucket <bucket\_name> | grep aws:SecureTransport | | |
| Reference | 2.1.2 Ensure S3 Bucket Policy is set to deny HTTP requests (Manual) | | |
| Site | https://aws.amazon.com/ko/premiumsupport/knowledge-center/s3-bucket-policy-for-config-rule/ https://aws.amazon.com/ko/blogs/security/how-to-use-bucket-policies-and-apply-defense-in-depth-to-help-secure-your-amazon-s3-data/ https://awscli.amazonaws.com/v2/documentation/api/latest/reference/s3api/get-bucket-policy.html https://docs.aws.amazon.com/AmazonS3/latest/userguide/security-best-practices.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-ssl-requests-only.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-s3-5 https://aws.amazon.com/ko/premiumsupport/knowledge-center/s3-bucket-policy-for-config-rule/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷의 MFA 삭제 활성화(3.1.3) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | Amazon S3 버킷에 MFA 삭제를 설정하는 것을 권장한다. | | |
| 설명 | MFA 삭제(MFA delete)를 활성화할 경우, 버킷의 버전 관리 상태가 변경되거나 객체 버전을 영구 삭제하는 요청이 들어왔을 때 추가적인 인증을 요구하여 실수로 버킷을 삭제하는 것을 방지할 수 있다. 삭제 작업을 요청한 사용자는 하드웨어 또는 가상 MFA 디바이스의 코드를 입력해야 하며, 이를 통해 보안 자격 증명이 손상된 경우에도 추가적인 인증을 통해 데이터를 보호할 수 있다.  ※ MFA 삭제는 버킷 소유자(루트 계정)만 활성화할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 버킷 버전 관리가 활성화된 S3 버킷에 "MFA 삭제(MFA delete)"가 설정되어 있는 경우   ▶ 취약 : 버킷 버전 관리가 활성화된 S3 버킷에 "MFA 삭제"가 설정되어 있지 않거나 버킷 버전 관리가 비활성화되어 있는 경우  ▶ Default Value : Disabled | | |
| 조치 방안 | ※ MFA 삭제는 Console에서는 설정할 수 없으며, CLI 또는 API를 통해 설정할 수 있다. ※ MFA 삭제는 버킷 소유자(루트 계정)만 활성화할 수 있다.  [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷의 버전 관리와 MFA 삭제(MFA delete) 기능을 활성화한다.  # aws s3api put-bucket-versioning --bucket <bucket name> --versioning-configuration Status=Enabled,MFADelete=Enabled --mfa "SERIAL <mfa serial number>" | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다. 4. 버킷 버전 관리(Bucket Versioning) 항목이 "활성화됨(Enabled)" 인지 확인한다.  5. Multi-Factor Authentication(MFA) 삭제(Multi-factor authentication (MFA) delete) 항목이 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷의 버전 관리와 MFA 삭제 기능이 활성화되어 있는지 확인한다.  # aws s3api get-bucket-versioning --bucket <bucket name>  아래와 같이 결과값이 출력된 경우, 버전 관리는 활성화되어 있지만 MFA 삭제는 비활성화된 상태이다.  {  "Status": "Enabled" } | | |
| Reference | 2.1.3 Ensure MFA Delete is enable on S3 buckets (Automated) | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/userguide/security-best-practices.html#mfa https://docs.aws.amazon.com/AmazonS3/latest/userguide/MultiFactorAuthenticationDelete.html https://docs.aws.amazon.com/cli/latest/reference/s3api/get-bucket-versioning.html https://docs.aws.amazon.com/cli/latest/reference/s3api/put-bucket-versioning.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Macie 활성화(3.1.4) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | Amazon Macie를 통해 S3 버킷 내 민감 데이터를 검색, 모니터링 및 분석하는 것을 권장한다. | | |
| 설명 | Amazon Macie는 머신러닝 및 패턴 분석을 사용하여 AWS 환경에서 민감 데이터를 검색, 모니터링 및 보호하는데 도움을 주는 관리형 데이터 보안 및 데이터 개인 정보보호 서비스이다. Macie를 통해 S3 버킷을 자동으로 평가 및 모니터링하여 조직에 지나치게 허용되거나 암호화되지 않은 버킷을 식별하고 보고할 수 있다. 잠재적인 보안 위험을 식별 및 조사하기 위해 S3 버킷의 퍼블릭 액세스 차단이 설정되어 있는지, 암호화가 되어있는지 등을 파악하고, 민감 데이터를 검색하여 보안 태세를 점검하는 것을 권장한다.  ※ Macie를 사용할 경우 추가적인 요금이 부과될 수 있다. | | |
| 진단 기준 | ▶ 양호 : Macie를 통해 모든 S3 버킷 내 민감 데이터를 검색하는 작업(job)이 설정되어 있는 경우   ▶ 취약 : Macier가 비활성화되어 있거나 Macie를 통해 모든 S3 버킷 내 민감 데이터를 검색하는 작업이 설정되어 있지 않은 경우    ▶ Default Value : Macie disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/macie/ 를 통해 Amazon Macie에 접속한다.  2. 좌측 사이드 바에서 '시작하기(Get started)'를 클릭한다.  3. "Macie 활성화(Enable Macie)" 버튼을 클릭한다.   ※ 민감 데이터가 발견될 경우 저장할 리포지토리를 생성한다. 4. 좌측 사이드 바에서 '설정(Settings)'의 '검색 결과(Discovery results)'를 클릭한다.  5. "지금 구성(Configure now)" 버튼을 클릭한다.  6. '민감한 데이터 검색 결과를 위한 리포지토리(Repository for sensitive data discovery results)' 에서 '새 버킷 생성(Create bucket)'을 선택한 후, 새 버킷 이름을 설정한다.  7. '고급(Advanced)'을 클릭하여, "모든 퍼블릭 액세스 차단(Block all public access)" 항목을 "예(Yes)"로 설정한다.  8. KMS 암호화 키를 선택한 후, 저장(Save) 버튼을 클릭한다.   ※ 민감 데이터를 검색하기 위한 작업을 생성한다. 9. 좌측 사이드 바에서 'S3 버킷(S3 buckets)'을 클릭한다.  10. Macie에서 분석할 버킷을 체크하고, 작업 생성(Create job) 버튼을 클릭한다.  11. 하단의 '빠른 생성(Quick create)' 버튼을 클릭한다.  12. 작업 이름과 작업 설명을 작성한 후, 다음(Next) 버튼을 클릭한다.  13. 제출(Submit) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/macie/ 를 통해 Amazon Macie에 접속한다.  2. 좌측 사이드 바에서 '결과(Findings)'의 '작업별(By job)'을 클릭한다.  3. S3 버킷의 작업별 결과가 있는지 확인한다. | | |
| Reference | 2.1.4 Ensure all data in Amazon S3 has been discovered, classified and secured when required. (Manual) | | |
| Site | https://aws.amazon.com/ko/macie/getting-started/ https://docs.aws.amazon.com/workspaces/latest/adminguide/data-protection.html https://docs.aws.amazon.com/macie/latest/user/data-classification.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 퍼블릭 액세스 차단 설정(3.1.5) | 위험도 | High |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | 의도하지 않은 사용자의 접근을 방지하기 위해 퍼블릭 액세스 차단 설정을 활성화하는 것을 권장한다. | | |
| 설명 | Amazon S3은 버킷 및 계정에 대한 퍼블릭 액세스 차단 기능을 통해 S3 리소스에 대한 퍼블릭 액세스를 관리할 수 있도록 지원한다. 퍼블릭 액세스 차단(버킷 설정)은 버킷 내 데이터의 의도하지 않은 우발적 또는 악의적 공개 노출을 방지하며, 퍼블릭 액세스 차단(계정 설정)은 각 AWS 계정의 모든 버킷(AWS 내 모든 리전)에 포함된 데이터의 공개 노출을 방지한다. 퍼블릭 액세스를 차단할 버킷의 범위는 데이터의 민감도, 최소 권한 및 사용 사례를 기반으로 결정해야 한다.  ※ 퍼블릭 액세스 차단(계정 설정)을 활성화할 경우, AWS 내 모든 리전에 적용된다. | | |
| 진단 기준 | ▶ 양호 : 퍼블릭 액세스 차단(Block Public Access settings for account)(버킷 설정, 계정 설정)이 활성화되어 있는 경우   ▶ 취약 : 퍼블릭 액세스 차단(버킷 설정, 계정 설정)이 비활성화되어 있는 경우   ▶ Default Value : 퍼블릭 액세스 차단(버킷 설정) = 활성화, 퍼블릭 액세스 차단(계정 설정) = 비활성화 | | |
| 조치 방안 | ※ 버킷 설정 [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다.  3. 권한(Permissoins) 탭을 클릭한다.  4. '퍼블릭 액세스 차단(버킷 설정)(Block public access (bucket settings))' 항목 우측의 '편집(Edit)' 버튼을 클릭한다.  5. "모든 퍼블릭 액세스 차단(Block all public access)"을 클릭한다. 6. 변경 사항 저장(Save changes) 버튼을 클릭한다.  ※ 민감정보를 포함하는 AWS 계정의 모든 버킷에 적용하는 것을 권장한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷들을 나열한다.  # aws s3 ls   2. 해당 버킷에 대해 모든 퍼블릭 액세스 차단을 설정한다.  # aws s3api put-public-access-block --bucket <name-of-bucket> --public-access-block-configuration "BlockPublicAcls=true, IgnorePublicAcls=true, BlockPublicPolicy=true, RestrictPublicBuckets=true"   ※ 계정 설정 [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 사이드 바에서 '이 계정의 퍼블릭 액세스 차단 설정(Block Public Access settings for this account)'을 클릭한다. 3. "이 계정의 퍼블릭 액세스 차단 설정" 항목의 '편집(Edit)' 버튼을 클릭한다.  4. "모든 퍼블릭 액세스 차단(Block all public access)"을 클릭한다.  5. 변경 사항 저장(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 해당 계정에 대해 모든 퍼블릭 액세스 차단을 설정한다. # aws s3control put-public-access-block --public-access-block-configuration "BlockPublicAcls=true, IgnorePublicAcls=true, BlockPublicPolicy=true, RestrictPublicBuckets=true" --account-id <account\_id> | | |
| 감사 방안 | ※ 버킷 설정 [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다.  3. 권한(Permissoins) 탭을 클릭한다.  4. '퍼블릭 액세스 차단(버킷 설정)(Block public access (bucket settings))' 항목이 각 버킷에 맞게 적절하게 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷 목록을 출력한다.  # aws s3 ls   2. 해당 버킷에서 퍼블릭 액세스 차단이 설정되어 있는지 확인한다.  # aws s3api get-public-access-block --bucket <name-of-the-bucket>   퍼블릭 액세스 차단이 활성화되어 있는 경우 다음과 같이 출력된다.  {  "PublicAccessBlockConfiguration": {  "BlockPublicAcls": true,  "IgnorePublicAcls": true,  "BlockPublicPolicy": true,  "RestrictPublicBuckets": true  }  }  ※ 계정 설정 [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 사이드 바에서 '이 계정의 퍼블릭 액세스 차단 설정(Block Public Access settings for this account)'을 클릭한다. 3. "이 계정의 퍼블릭 액세스 차단 설정" 항목이 각 AWS 계정에 맞게 적절하게 설정되어 있는지 확인한다.   [AWS CLI] 1. 해당 계정에 대한 퍼블릭 액세스 차단 설정을 확인한다.  # aws s3control get-public-access-block --account-id <account\_id>  퍼블릭 액세스 차단이 활성화되어 있는 경우 다음과 같이 출력된다.  {  "PublicAccessBlockConfiguration": {  "IgnorePublicAcls": true,  "BlockPublicPolicy": true,  "BlockPublicAcls": true,  "RestrictPublicBuckets": true  } } | | |
| Reference | 2.1.5 Ensure that S3 Buckets are configured with 'Block public access (bucket settings)' (Automated) | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/user-guide/block-public-access-account.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 태그 지정(3.1.6) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | S3 버킷에 태그를 지정할 경우, 운영 측면에서 관리하기에 더 용이하다. | | |
| 설명 | 태그를 지정할 경우 용도, 관리자, 만료 날짜 등 AWS 리소스를 다양한 방식으로 분류할 수 있다. 이 기능은 비슷한 유형의 리소스가 많을 때 유용하며, 지정한 태그에 따라 특정 리소스를 빠르게 식별 및 추적할 수 있다. ※ 사용자 지정 태그 식별자는 대/소문자를 구분한다. | | |
| 진단 기준 | ▶ 양호 : 각 S3 버킷에 태그가 설정되어 있는 경우   ▶ 취약 : 각 S3 버킷에 태그가 설정되어 있지 않는 경우   ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다.  4. "태그(Tags)" 의 '편집(Edit)' 버튼을 클릭한다.  5. 태그 추가(Add tag) 를 클릭한다.  6. 키와 값을 입력한 후, 변경 사항 저장(Save changes) 을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다.  4. "태그(Tags)"가 존재하는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/general/latest/gr/aws\_tagging.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 버전 관리 설정(3.1.7) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | 민감 정보가 있는 버킷의 경우, Amazon S3 버킷에 저장된 모든 버전의 모든 객체를 보존, 검색 및 복원을 할 수 있도록 버전 관리를 활성화하는 것을 권장한다. | | |
| 설명 | 버킷 버전 관리(Bucket Versioning) 기능을 활성화할 경우, Amazon S3 버킷에 저장된 모든 버전의 모든 객체를 보존 및 검색할 수 있다. 또한 의도치 않은 사용자 작업 및 애플리케이션 장애로부터 쉽게 복구할 수 있으며, Amazon S3가 동일한 객체에 대해 여러 쓰기 요청을 동시에 수신하는 경우 모든 객체를 저장한다. | | |
| 진단 기준 | ▶ 양호 : 각 S3 버킷에 "버킷 버전 관리(Bucket Versioning)"가 활성화되어 있는 경우   ▶ 취약 : 각 S3 버킷에 "버킷 버전 관리"가 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다.  4. "버킷 버전 관리(Bucket Versioning)" 의 '편집(Edit)' 버튼을 클릭한다.  5. 버킷 버전 관리 항목에서 "활성화(Enable)"를 선택한다.  6. 변경 사항 저장(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷의 버전 관리를 활성화한다.  # aws s3api put-bucket-versioning --bucket <bucket name> --versioning-configuration Status=Enabled | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다.  4. 버킷 버전 관리(Bucket Versioning) 항목이 "활성화됨(Enabled)" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷의 버전 관리가 활성화되어 있는지 확인한다.  # aws s3api get-bucket-versioning --bucket <bucket name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonS3/latest/dev/Versioning.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷의 객체 수준 로깅 이벤트 쓰기 설정(3.1.8) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | 추적을 저장하는 S3 버킷에 대한 객체 수준 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | CloudTrail은 GetObject, DeleteObject, PutObject 같은 Amazon S3 객체 수준 API 작업 로깅을 지원한다. 객체 수준 로깅을 사용할 경우, 조직 내의 데이터 규정 준수 요구사항 충족 및 포괄적인 보안 분석을 수행할 수 있다. 또한, AWS 계정의 특정 사용자 동작 패턴을 모니터링하고, Amazon CloudWatch 이벤트를 사용하여 S3 버킷 내의 객체 수준 API 활동에 대한 즉각적인 조치를 취할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 추적 버킷에 대해 'AWS CloudTrail 데이터 이벤트(AWS CloudTrail data events)'의 "쓰기(Write)" 로깅이 활성화되어 있는 경우    ▶ 취약 : 추적 버킷에 대해 'AWS CloudTrail 데이터 이벤트'의 "쓰기" 로깅이 비활성화되어 있는 경우    ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 S3 버킷을 클릭한다.  3. 속성(Properties) 탭을 클릭한다.  4. AWS CloudTrail 데이터 이벤트(AWS CloudTrail data events)의 "CloudTrail에서 구성(Configure in CloudTrail)" 버튼을 클릭한다.  5. 각 추적을 클릭한다.  6. 데이터 이벤트(Data events) 항목의 '편집(Edit)' 버튼을 클릭한다.  7. 이벤트 유형(Event type) 에서 "데이터 이벤트(Data events)"를 체크한다.  8. '현재 및 향후 모든 S3 버킷(All current and future S3 buckets)' 항목 또는 '개별 버킷 선택(Individual bucket selection)'에서 추적 로그 위치의 버킷을 선택한 후 "쓰기(Write)"를 체크한다.  9. 변경 사항 저장(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 추적 로그를 저장하고 있는 S3 버킷에 대해 데이터 이벤트의 "쓰기" 로깅을 활성화한다. # aws cloudtrail put-event-selectors --trail-name <trail name> --event-selectors '[{ "ReadWriteType": "WriteOnly", "IncludeManagementEvents":true, "DataResources": [{ "Type": "AWS::S3::Object", "Values": ["<s3 bucket arn>/"] }] }]'   ※ 모든 리전의 추적에 대해 수행한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 각 추적을 클릭한다.  4. '데이터 이벤트(Data events)' 항목에서 해당 추적의 로그가 저장되는 S3 버킷 또는 "현재 및 향후 모든 S3 버킷(All current and future S3 buckets)"에 "쓰기(Write)"가 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 추적의 로그가 저장되는 S3 버킷("S3BucketName")을 확인한다.  # aws cloudtrail get-trail --name <trail name>  2. 해당 버킷에 대해 데이터 이벤트의 "쓰기" 로깅이 활성화되어 있는지 확인한다. "ReadWriteType" 항목이 "WriteOnly" 또는 "All" 이고, "DataResources"의 "Values"는 1번에서 확인한 S3 버킷의 ARN이거나 "arn:aws:s3" 이어야 한다.  # aws cloudtrail get-event-selectors --trail-name <trail name>  예시는 다음과 같다.  {  "TrailARN": "<trail arn>",  "EventSelectors": [  {  "ReadWriteType": "WriteOnly",  "IncludeManagementEvents": true,  "DataResources": [  {  "Type": "AWS::S3::Object",  "Values": [  "<s3 bucket arn>"  ]  }  ],  "ExcludeManagementEventSources": []  }  ] }  ※ 모든 리전의 추적에 대해 수행한다. | | |
| Reference | 3.10 Ensure that Object-level logging for write events is enabled for S3 bucket (Automated) | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/user-guide/what-is-s3.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷의 객체 수준 로깅 이벤트 읽기 설정(3.1.9) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | 추적을 저장하는 S3 버킷에 대한 객체 수준 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | CloudTrail은 GetObject, DeleteObject, PutObject 같은 Amazon S3 객체 수준 API 작업 로깅을 지원한다. 객체 수준 로깅을 사용할 경우, 조직 내의 데이터 규정 준수 요구사항 충족 및 포괄적인 보안 분석을 수행할 수 있다. 또한, AWS 계정의 특정 사용자 동작 패턴을 모니터링하고, Amazon CloudWatch 이벤트를 사용하여 S3 버킷 내의 객체 수준 API 활동에 대한 즉각적인 조치를 취할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 추적 버킷에 대해 'AWS CloudTrail 데이터 이벤트(AWS CloudTrail data events)'의 "읽기(Read)" 로깅이 활성화되어 있는 경우    ▶ 취약 : 추적 버킷에 대해 'AWS CloudTrail 데이터 이벤트'의 "읽기" 로깅이 비활성화되어 있는 경우    ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 S3 버킷을 클릭한다.  3. 속성(Properties) 탭을 클릭한다.  4. AWS CloudTrail 데이터 이벤트(AWS CloudTrail data events)의 "CloudTrail에서 구성(Configure in CloudTrail)" 버튼을 클릭한다.  5. 각 추적을 클릭한다.  6. 데이터 이벤트(Data events) 항목의 '편집(Edit)' 버튼을 클릭한다.  7. 이벤트 유형(Event type) 에서 "데이터 이벤트(Data events)"를 체크한다.  8. '현재 및 향후 모든 S3 버킷(All current and future S3 buckets)' 항목 또는 '개별 버킷 선택(Individual bucket selection)'에서 추적 로그 위치의 버킷을 선택한 후 "읽기(Read)"를 체크한다.  9. 변경 사항 저장(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 추적 로그를 저장하고 있는 S3 버킷에 대해 데이터 이벤트의 "읽기" 로깅을 활성화한다. # aws cloudtrail put-event-selectors --trail-name <trail name> --event-selectors '[{ "ReadWriteType": "ReadOnly", "IncludeManagementEvents":true, "DataResources": [{ "Type": "AWS::S3::Object", "Values": ["<s3 bucket arn>/"] }] }]'   ※ 모든 리전의 추적에 대해 수행한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 각 추적을 클릭한다.  4. '데이터 이벤트(Data events)' 항목에서 해당 추적의 로그가 저장되는 S3 버킷 또는 "현재 및 향후 모든 S3 버킷(All current and future S3 buckets)"에 "읽기(Read)"가 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 추적의 로그가 저장되는 S3 버킷("S3BucketName")을 확인한다.  # aws cloudtrail get-trail --name <trail name>  2. 해당 버킷에 대해 데이터 이벤트의 "읽기" 로깅이 활성화되어 있는지 확인한다. "ReadWriteType" 항목이 "ReadOnly" 또는 "All" 이고, "DataResources"의 "Values"는 1번에서 확인한 S3 버킷의 ARN이거나 "arn:aws:s3" 이어야 한다.  # aws cloudtrail get-event-selectors --trail-name <trail name>  예시는 다음과 같다.  {  "TrailARN": "<trail arn>",  "EventSelectors": [  {  "ReadWriteType": "ReadOnly",  "IncludeManagementEvents": true,  "DataResources": [  {  "Type": "AWS::S3::Object",  "Values": [  "<s3 bucket arn>"  ]  }  ],  "ExcludeManagementEventSources": []  }  ] }  ※ 모든 리전의 추적에 대해 수행한다. | | |
| Reference | 3.11 Ensure that Object-level logging for read events is enabled for S3 bucket (Automated) | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/user-guide/enable-cloudtrail-events.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 객체 잠금 활성화(3.1.10) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | Amazon S3 버킷에 객체 잠금이 활성화되어 있는지 확인하는 것을 권장한다. | | |
| 설명 | 객체 잠금 기능은 WORM(Write-Once-Read-Many) 모델을 사용하여 객체를 저장하며, 일정 시간 또는 무기한으로 객체가 삭제되거나 덮어쓰기 되는 것을 방지할 수 있다. S3 버킷의 객체 잠금을 활성화할 경우, 버킷 내 객체가 영구적으로 잠길 수 있으며 해당 버킷과 객체는 삭제할 수 없다. 따라서 데이터 무결성 및 규제 준수를 위해 객체를 삭제해서는 안되는 경우에만 객체 잠금을 활성화하는 것을 권장한다.  ※ 객체 잠금은 새 버킷을 생성할 때만 적용할 수 있다. 기존 버킷에 객체 잠금을 활성화하기 위해서는 AWS Support에 문의해야 한다.  ※ 객체 잠금을 활성화한 버킷은 버전 관리가 자동으로 활성화되며, 서버 액세스 로깅을 활성화할 수 없다.  ※ 객체 잠금을 활성화할 경우, 다시 객체 잠금을 비활성화하거나 버전 관리를 일시중지 할 수 없다. | | |
| 진단 기준 | ▶ 양호 : 각 S3 버킷에 "객체 잠금(Object lock)"이 활성화되어 있는 경우   ▶ 취약 : 각 S3 버킷에 "객체 잠금"이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | ※ 객체 잠금은 새 버킷을 생성할 때만 적용할 수 있다. 기존 버킷에 객체 잠금을 활성화하기 위해서는 AWS Support에 문의해야 한다.   [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 버킷 만들기(Create bucket) 버튼을 클릭한다. 3. 버킷 이름, 리전 등을 설정한 후, '고급 설정(Advanced settings)'을 클릭한다.  4. "객체 잠금(Object Lock)" 항목에서 "활성화(Enable)"를 선택한다. | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 속성(Properties) 탭을 선택한다.  4. 객체 잠금(Object Lock) 항목이 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷에 객체 잠금이 활성화되어 있는지 확인한다.  # aws s3api get-object-lock-configuration --bucket <bucket name>  예시는 다음과 같다.  {  "ObjectLockConfiguration": {  "ObjectLockEnabled": "Enabled"  } } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/userguide/object-lock.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-default-lock-enabled.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 정책 설정(3.1.11) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | S3 버킷 정책을 설정하여 버킷 내 객체에 대한 액세스를 관리하는 것을 권장한다. | | |
| 설명 | 버킷 정책은 리소스 기반의 AWS IAM(Identity and Access Management) 정책으로 다른 AWS 계정이나 IAM 사용자에게 버킷과 버킷에 포함된 객체에 대한 액세스 권한을 부여할 수 있다. 최소 권한의 원칙에 따라 버킷에 액세스할 수 있는 사용자 또는 그룹을 명시적으로 지정하는 것을 권장한다.  ※ 버킷 정책의 최대 크기는 20KB 이다. | | |
| 진단 기준 | ▶ 양호 : 각 S3 버킷에 버킷 정책(Bucket policy)이 설정되어 있으며 버킷 정책에 따라 액세스가 관리되는 경우    ▶ 취약 : 각 S3 버킷에 버킷 정책이 설정되어 있지 않거나 버킷 정책에 따라 액세스가 관리되지 않는 경우   ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 권한(Permissions) 탭을 선택한다.  4. 버킷 정책(Bucket policy) 의 '편집(Edit)' 버튼을 클릭한다.  5. 정책 생성기(Policy generator) 버튼을 클릭한다.  6. Step1. Select Policy Type 에서 "S3 Bucket Policy"를 선택한다.  7. Step2. Add Statement(s) 에서 Allow 또는 Deny 를 적용할 Principal, Actions, ARN(Amazon Resource Name) 등을 설정한다.  8. 'Add Statement' 버튼을 클릭한다.  9. 정책 구문을 확인한 후, 'Generate Policy' 버튼을 클릭한다.  10. 생성된 정책 json 구문을 복사하여, S3 버킷 정책에 붙여 넣는다.  11. '변경 사항 저장(Save changes)' 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 권한(Permissions) 탭을 선택한다.  4. 버킷 정책(Bucket policy)이 설정되어 있는지 확인한다. 또한, 버킷 정책에 따라 액세스가 관리되고 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 S3 버킷에 버킷 정책이 설정되어 있는지 확인한다.  # aws s3api get-bucket-policy --bucket <bucket name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonS3/latest/userguide/bucket-policies.html https://docs.aws.amazon.com/ko\_kr/AmazonS3/latest/userguide/example-bucket-policies.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-grantee-check.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 교차 리전 복제 활성화(3.1.12) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | S3 | | |
| 요약 | Amazon S3 버킷에 교차 리전 복제를 활성화하는 것을 권장한다. | | |
| 설명 | Amazon S3는 여러 지역에 있는 다양한 리전에 데이터를 저장할 수 있으며, 객체 복제 기능을 사용할 경우 S3 버킷 내 객체를 비동기식으로 자동 복사할 수 있다. 또한, 동일한 AWS 계정 또는 다른 계정, 단일 대상 버킷 또는 여러 대상 버킷에 복제할 수 있다. 교차 리전 복제(Cross-Region Replication)를 활성화할 경우, 규정 준수 요구 사항에 따라 지리적으로 떨어진 서로 다른 리전 간에 데이터를 복제할 수 있으며 사용자와 지리적으로 더 가까운 AWS 리전에 객체 복사본을 유지하여 객체 액세스 지연 시간을 최소화할 수 있다.  ※ 교차 리전 복제를 사용할 경우, 원본 및 대상 S3 버킷 모두 "버킷 버전 관리"가 활성화되어 있어야 한다.  ※ 객체 잠금이 활성화된 경우, 교차 리전 복제를 활성화하기 위해서는 AWS Support에 문의해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 S3 버킷에 "교차 리전 복제(Cross-Region Replication)"가 활성화되어 있는 경우   ▶ 취약 : 각 S3 버킷에 "교차 리전 복제"가 활성화되어 있지 않은 경우  ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 관리(Management) 탭을 클릭한다.  4. 복제 규칙(Replication rules) 항목의 '복제 규칙 생성(Create replication rule)' 버튼을 클릭한다.  5. 복제 규칙 이름, 상태, 규칙 범위, IAM 역할 등을 설정한 후, 대상(Destination) 항목에서 해당 버킷이 저장된 리전이 아닌 다른 리전의 버킷을 선택한다. 6. 저장(Save) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 각 버킷을 클릭한다. 3. 관리(Management) 탭을 클릭한다.  4. 복제 규칙(Replication rules)에서 '대상 리전(Destination Region)'이 해당 버킷이 저장된 리전이 아닌 다른 리전으로 설정되고, '상태(Status)'가 "활성화(Enabled)"인 규칙이 존재하는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 S3 버킷에 교차 리전 복제가 활성화되어 있는지 확인한다.  # aws s3api get-object-lock-configuration --bucket <bucket name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/userguide/security-best-practices.html#cross-region https://docs.aws.amazon.com/AmazonS3/latest/userguide/replication.html#crr-scenario https://docs.aws.amazon.com/ko\_kr/AmazonS3/latest/userguide/replication-walkthrough1.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EBS 볼륨 암호화 설정(3.2.1) | 위험도 | High |
| 대분류 | Storage | | |
| 중분류 | EBS | | |
| 요약 | Amazon EBS에 데이터 보관 시 암호화하는 것을 권장한다. | | |
| 설명 | EBS(Elastic Block Store)에 암호화를 적용할 경우, 내부 데이터가 의도하지 않게 노출될 가능성이 낮아지며 유출되었더라도 복호화되지 않는다면 영향을 무효화할 수 있다. 기본적으로 EBS 볼륨 암호화는 새로 생성된 EBS 볼륨에만 적용되며 기존 EBS 볼륨에는 적용되지 않는다. | | |
| 진단 기준 | ▶ 양호 : 모든 EBS 볼륨이 암호화되어 있는 경우   ▶ 취약 : EBS 볼륨이 암호화되어 있지 않은 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 새 EBS 생성 시 암호화 적용 설정 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 계정 속성(Account attributes)에서 'EBS 암호화(EBS encryption)'를 클릭한다.  3. EBS 암호화(EBS encryption)의 '관리(Manage)'를 클릭한다.  4. "새 EBS 볼륨이 항상 암호화(Always encrypt new EBS volumes)" 항목에서 "활성화(Enable)"를 선택한다.  5. EBS 암호화 업데이트(Update EBS encryption) 버튼을 클릭한다.  6. 새 인스턴스 생성시 스토리지 내 루트 디바이스 유형이 EBS로 활성화되어 있는지 확인한다.  ※ 위 설정시 기존에 생성된 EBS는 암호화되지 않는다. 기존에 생성된 EBS를 암호화하는 방법은 현재 지원되지 않는다. 현재 운영중인 인스턴스에 연결된 EBS에 암호화를 적용하기 위해서는 암호화된 EBS를 생성하여 기존 EBS를 해제한 후, 새로 생성된 EBS를 연결해야 한다. 단, 이를 위해서는 인스턴스를 재부팅 해야 한다.   [AWS CLI] 1. 다음 명령어를 실행하여 "EbsEncryptionByDefault" 항목을 "true" 로 설정한다.  # aws --region <region> ec2 get-ebs-encryption-by-default  ※ 각 리전마다 설정해야 한다. | | |
| 감사 방안 | [AWS Console] ※ 새 EBS 생성 시 암호화 적용 여부 확인 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 우측의 계정 속성(Account attributes)에서 'EBS 암호화(EBS encryption)'를 클릭한다.  3. "새 EBS 볼륨이 항상 암호화(Always encrypt new EBS volumes)" 항목이 "활성화(Enabled)" 되어 있는지 확인한다. ※ 각 리전마다 2~4번을 반복하여 확인한다.   ※ 현재 생성된 EBS 암호화 여부 확인 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 'Elastic Block Store'의 '볼륨(Volumes)'을 클릭한다. 3. 태그 및 속성별 필터 또는 키워드별 검색바에 "암호화 : 암호화되지 않음 (Encryption : Not Encrypted)" 필터를 추가한다. 4. 암호화가 되지 않은 볼륨이 없는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 "EbsEncryptionByDefault" 항목이 "true" 인지 확인한다. # aws --region <region> ec2 get-ebs-encryption-by-default  ※ 각 리전마다 확인해야 한다. | | |
| Reference | 2.2.1 Ensure EBS volume encryption is enabled (Manual) | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSEncryption.html https://aws.amazon.com/ko/blogs/aws/new-opt-in-to-default-encryption-for-new-ebs-volumes/ https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/encrypted-volumes.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-ec2-3 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EBS 스냅샷 Private 설정(3.2.2) | 위험도 | High |
| 대분류 | Storage | | |
| 중분류 | EBS | | |
| 요약 | 모든 EBS 스냅샷은 Private으로 설정 및 암호화하는 것을 권장한다. | | |
| 설명 | EBS(Elastic Block Store)는 AWS EC2에 연결할 수 있는 영구 블록 스토리지 볼륨으로 EBS 스냅샷은 특정 시점에 EBS 볼륨을 백업하거나 이전 상태로 복원할 때 사용된다. EBS 스냅샷이 외부에 노출될 경우, 모든 사용자가 해당 스냅샷에 접근 및 인스턴스를 생성하여 민감 정보에 접근할 수 있으므로 Private으로 설정 및 암호화하는 것을 권장한다.  Public : 모든 AWS 계정이 해당 스냅샷을 통해 볼륨 생성 가능 ※ public으로 설정 시 group이 all로 표시된다.  Private - explicit : 스냅샷 소유자가 지정한 AWS 계정만 해당 스냅샷을 통해 볼륨 생성 가능 - implicit : AWS 계정이 소유한 해당 스냅샷을 통해 접근 가능 | | |
| 진단 기준 | ▶ 양호 : EBS의 스냅샷이 "Private"으로 설정되어 있는 경우   ▶ 취약 : EBS의 스냅샷이 "Public"으로 설정되어 있는 경우   ▶ Default Value : Private | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 'Elastic Block Store'의 '스냅샷(Snapshots)'을 클릭한다. 3. 해당 스냅샷을 체크한 후, 상단의 '작업(Actions)' 버튼을 클릭하여 '권한 수정(Modifiy Permissions)'을 선택한다. 4. "이 스냅샷의 현재 상태(This snapshot is currently)" 항목을 "프라이빗(Private)" 으로 선택한 후, 저장(Save) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 'Elastic Block Store'의 '스냅샷(Snapshots)'을 클릭한다. 3. 각 스냅샷을 체크한 후, 상단의 '작업(Actions)' 버튼을 클릭하여 '권한 수정(Modifiy Permissions)'을 선택한다.  4. "이 스냅샷의 현재 상태(This snapshot is currently)" 항목이 "프라이빗(Private)" 인지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AWSEC2/latest/UserGuide/EBSSnapshots.html https://docs.aws.amazon.com/ko\_kr/AWSEC2/latest/UserGuide/ebs-modifying-snapshot-permissions.html https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API\_DescribeSnapshots.html  https://docs.aws.amazon.com/config/latest/developerguide/ebs-snapshot-public-restorable-check.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-ec2-1 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EBS 백업 계획 설정(3.2.4) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | EBS | | |
| 요약 | AWS Backup 백업 계획을 통해 EBS 볼륨을 정기적으로 백업하고, 일정 기간 이상 보존하는 것을 권장한다. | | |
| 설명 | AWS Backup은 클라우드 및 온프레미스의 AWS 서비스 전반에서 데이터 백업을 쉽게 중앙집중화 및 자동화할 수 있는 서비스이며, 백업 관리를 간소화하고 비즈니스 및 규제 백업 규정 준수 요구사항을 충족할 수 있다. 백업 계획을 통해 백업 일정, 백업 보존 규칙 및 수명 주기 규칙을 비롯한 백업 요구 사항을 정의하여, 해당 계획에 따라 리소스를 자동으로 백업하고 유지할 수 있다. 기본 백업 기간은 오전 5시(UTC)에 시작하여 8시간 동안 지속된다. 매일(24시간마다) 백업하도록 설정하는 것을 권장한다.   백업 계획을 생성하는 방법은 세 가지가 있다.  - 템플릿으로 시작 : AWS Backup에서 제공하는 템플릿을 기반으로 백업 계획을 생성할 수 있다.   - Daily-35day-Retention (default) : 매일 백업 생성, 35일간 보존  - Daily-Monthly-1yr-Retention : 매일 및 매달 백업 생성, 1년간 보존  - Daily-Weekly-Monthly-5yr-Retention : 매일, 매주, 매달 백업 생성, 5년간 보존  - Daily-Weekly-Monthly-7yr-Retention : 매일, 매주, 매달 백업 생성, 7년간 보존 - 새 계획 수립 : 처음부터 새 백업 계획을 구성할 수 있다.  - JSON을 사용하여 계획 정의 : 기존 백업 계획의 JSON 표현식을 수정하거나 새 표현식을 생성할 수 있다. | | |
| 진단 기준 | ▶ 양호 : AWS Backup 백업 계획에 각 EBS 파일 시스템이 할당되어 정기적으로 백업이 수행되고 있는 경우   ▶ 취약 : AWS Backup 백업 계획에 각 EBS 파일 시스템이 할당되어 있지 않은 경우   ▶ Default Value : - ※ 리소스 자체 백업 기능을 통해 리소스를 백업하고 있는 경우, 해당 항목은 예외처리 할 수 있다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 백업 계획 생성(Create Backup plan) 버튼을 클릭한다.  4. 시작 옵션(Start options)에서 '템플릿으로 시작(Start with a template)'을 선택한다.  5. 템플릿 선택(Choose template)에서 템플릿 계획을 선택한다. 모든 템플릿에는 매일 백업하는 규칙(DailyBackups)이 적용되어 있다.  6. 백업 계획의 이름을 설정한 후, 계획 생성(Create plan) 버튼을 클릭한다.  ※ 시작 옵션에서 '새 계획 수립(Build a new plan)' 또는 'JSON을 사용하여 계획 정의(Define a plan using JSON)'를 선택하여 사용자가 직접 백업 계획을 구성할 수 있으며, 백업 빈도를 "매일(Daily)"로 설정하는 것을 권장한다. 7. 해당 백업 계획으로 이동한다.  8. 리소스 할당(Resource assignments)에서 "리소스 할당(Assign resources)" 버튼을 클릭한다.  9. 리소스 할당 이름 및 IAM 역할을 지정한 후, "다음을 기준으로 할당(Assign by)"에서 "리소스 ID(Resource ID)"를 선택한다.  10. "리소스 유형(Resource type)"에서 "EBS"를 선택하고, "볼륨 ID(Volume ID)"를 선택한다.  11. 리소스 할당(Assign resources) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 백업 계획을 생성한다.  # aws backup create-backup-plan --backup-plan <value>  예시는 다음과 같다.  # aws backup create-backup-plan --backup-plan "{\"BackupPlanName\":\"Example-Backup-Plan\",\"Rules\":[{\"RuleName\":\"DailyBackups\",\"ScheduleExpression\":\"cron(0 5 ? \* \* \*)\",\"StartWindowMinutes\":480,\"TargetBackupVaultName\":\"Default\",\"Lifecycle\":{\"DeleteAfterDays\":35}}]}"  2. 해당 백업 계획에 리소스를 할당한다.  # aws backup create-backup-selection --backup-plan-id <backup-plan-id> --backup-selection SelectionName="string",IamRoleArn="<iam role arn>",Resources="<resource arn>","<resource arn>",ListOfTags=[{ConditionType=string,ConditionKey=string,ConditionValue=string},{ConditionType=string,ConditionKey=string,ConditionValue=string}] | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 각 백업 계획을 클릭한다.  4. 백업 규칙(Backup rules)이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다.  5. 리소스 할당(Resource assignments)에 EBS 볼륨이 지정되어 있는지 확인한다.  ※ 각 EBS 볼륨마다 설정된 백업 계획 및 백업 규칙은 다를 수 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 백업 계획의 백업 규칙이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다. # aws backup get-backup-plan --backup-plan-id <backup-plan-id>  2. 백업 계획에 각 EBS 볼륨이 할당되어 있는지 확인한다.  # aws backup list-backup-selections --backup-plan-id <backup-plan-id> # aws backup get-backup-selection --backup-plan-id <backup-plan-id> --selection-id <selection id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/aws-backup/latest/devguide/about-backup-plans.html https://docs.aws.amazon.com/aws-backup/latest/devguide/create-a-scheduled-backup.html https://docs.aws.amazon.com/config/latest/developerguide/ebs-in-backup-plan.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/cli/latest/reference/backup/create-backup-plan.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EBS 삭제 대기 상태 설정(3.2.5) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | EBS | | |
| 요약 | 인스턴스 종료 시 EBS 볼륨이 삭제 대기 상태가 되도록 설정하는 것을 권장한다. | | |
| 설명 | Amazon EBS(Elastic Block Store)는 EC2 인스턴스에 연결할 수 있는 블록 수준의 스토리지 볼륨이다. 인스턴스에 볼륨을 연결할 경우, 물리적 하드 드라이브처럼 사용할 수 있으며 인스턴스의 수명과 관계없이 지속적으로 유지할 수 있다. 또한, 단일 하드웨어 구성 요소의 장애로 인한 데이터 손실을 방지하기 위해 해당 가용 영역 내에서 자동으로 복제된다. EBS 볼륨은 EC2 인스턴스의 실행 주기와는 독립적으로 유지되며, 인스턴스가 종료될 때 EBS 볼륨을 자동으로 삭제하도록 선택할 수 있다. 사용자의 실수로 인스턴스가 종료되는 상황에 대비하여, 인스턴스 종료 시 복구를 위해 EBS 볼륨을 즉시 삭제하지 않고 유지하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 EC2 인스턴스에 연결된 EBS 볼륨의 "DeleteOnTermination" 속성이 "false"로 설정되어 있는 경우  ▶ 취약 : 각 EC2 인스턴스에 연결된 EBS 볼륨의 "DeleteOnTermination" 속성이 "true"로 설정되어 있는 경우  ▶ Default Value : 루트 볼륨의 경우 true, 루트 외 볼륨의 경우 false ※ 인스턴스에 연결되지 않은 EBS 볼륨의 경우, 예외처리된다. | | |
| 조치 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 EC2 인스턴스에 연결된 EBS 볼륨의 "DeleteOnTermination" 속성을 "false"로 설정한다.  # aws ec2 modify-instance-attribute --instance-id <instance id> --block-device-mappings "[{\"DeviceName\": \"<deviceName>\",\"Ebs\":{\"DeleteOnTermination\":false}}]" ※ DeviceName은 루트 볼륨의 디바이스 이름이며, 루트 외 볼륨의 경우 다른 디바이스 이름을 지정하여야 한다. | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 EC2 인스턴스에 연결된 EBS 볼륨의 "DeleteOnTermination" 속성이 "false" 로 설정되어 있는지 확인한다.  # aws ec2 describe-instance-attribute --instance-id <instance id> --attribute blockDeviceMapping | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AWSEC2/latest/UserGuide/terminating-instances.html#preserving-volumes-on-termination https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/ec2-volume-inuse-check.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EFS 자동 백업 설정(3.3.1) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | EFS | | |
| 요약 | EFS 파일 시스템에 자동 백업을 설정하는 것을 권장한다. | | |
| 설명 | Amazon EFS(Elastic File System) 파일 시스템 내 데이터를 보호하기 위하여 AWS Backup을 통해 자동으로 백업하는 것을 권장한다. AWS Backup은 클라우드 및 온프레미스의 AWS 서비스 전반에서 데이터 백업을 쉽게 중앙집중화 및 자동화할 수 있는 서비스이며, 백업 관리를 간소화하고 비즈니스 및 규제 백업 규정 준수 요구사항을 충족할 수 있다. 리소스에서 Backup 계획을 설정할 경우, 해당 계획에 따라 리소스를 자동으로 백업하고 유지할 수 있다.  ※ 백업 프로세스가 진행되는 동안에는 파일 시스템을 수정하는 애플리케이션 또는 프로세스를 일시 중지하는 것을 권장한다. 또는 파일 시스템이 수정되지 않는 기간에 백업하도록 예약하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 EFS 파일 시스템의 자동 백업(Automatic backups)이 활성화되어 있는 경우   ▶ 취약 : 각 EFS 파일 시스템의 자동 백업이 활성화되어 있지 않은 경우   ▶ Default Value : Enabled | | |
| 조치 방안 | [AWS Console] 1. https://ap-northeast-2.console.aws.amazon.com/efs/ 를 통해 Amazon Elastic File System에 접속한다. 2. 해당 EFS 파일 시스템을 클릭한다.  3. 우측 상단의 '편집(Edit)' 버튼을 클릭한다.  4. 자동 백업(Automatic backups) 항목의 "자동 백업 활성화(Enable automatic backups)"를 선택한다.  5. 변경 사항 저장(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 EFS 파일 시스템의 자동 백업을 활성화한다.  # aws efs put-backup-policy --file-system-id <efs id> --backup-policy Status="ENABLED" | | |
| 감사 방안 | [AWS Console] 1. https://ap-northeast-2.console.aws.amazon.com/efs/ 를 통해 Amazon Elastic File System에 접속한다. 2. 각 EFS 파일 시스템을 클릭한다.  3. 자동 백업(Automatic backups) 항목이 "활성화됨(Enabled)" 으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 EFS 파일 시스템의 "BackupPolicy" 항목이 "ENABLED" 로 설정되어 있는지 확인한다.  # aws efs describe-backup-policy --file-system-id <efs id>  예시는 다음과 같다.  {  "BackupPolicy": {  "Status": "ENABLED"  } } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/efs/latest/ug/efs-backup-solutions.html https://docs.aws.amazon.com/efs/latest/ug/awsbackup.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EFS 암호화 설정(3.3.2) | 위험도 | Low |
| 대분류 | Storage | | |
| 중분류 | EFS | | |
| 요약 | EFS 파일 시스템의 암호화를 활성화하는 것을 권장한다. | | |
| 설명 | EFS 파일 시스템의 유휴 시 데이터 암호화를 활성화할 경우, Amazon EFS용 AWS 관리형 CMK(aws/elasticfilesystem)를 통해 파일 시스템 내 유휴 데이터 및 메타 데이터(파일 이름, 디렉터리 이름, 디렉터리 내용 등)을 암호화한다. 파일 시스템 내 민감 데이터를 보호하기 위하여 암호화를 설정하는 것을 권장한다.  ※ 고객 관리형 CMK로 암호화할 경우, 사용자가 유휴 암호화 파일 내용에 액세스하기 위해서는 해당 키가 활성화된 상태여야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 EFS 파일 시스템의 암호화가 활성화되어 있는 경우   ▶ 취약 : 각 EFS 파일 시스템의 암호화가 비활성화되어 있는 경우   ▶ Default Value : Enabled(Amazon EFS용 AWS 관리형 CMK(aws/elasticfilesystem)) | | |
| 조치 방안 | ※ 새 EFS 파일 시스템을 생성할 경우 [AWS Console] 1. https://ap-northeast-2.console.aws.amazon.com/efs/ 를 통해 Amazon Elastic File System에 접속한다. 2. 파일 시스템 생성(Create file system) 버튼을 클릭한다.  3. 사용자 지정(Customize) 버튼을 클릭한다.  4. '유휴 시 데이터 암호화 활성화(Enable encryption of data at rest)'를 체크한다.  5. 다음(Next) 버튼을 클릭한다.  6. 네트워크 액세스, 파일 시스템 정책 등을 설정한 후, 생성(Create) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://ap-northeast-2.console.aws.amazon.com/efs/ 를 통해 Amazon Elastic File System에 접속한다. 2. 각 EFS 파일 시스템을 클릭한다.  3. 암호화됨(Encrypted) 항목이 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EFS 파일 시스템에 암호화가 설정되어 있는지 확인한다. "Encrypted" 항목이 "true"로 설정되어 있어야 한다.  # aws efs describe-file-systems --file-system-id <file system id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/efs/latest/ug/encryption-at-rest.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/efs-encrypted-check.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-efs-1 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EFS 백업 계획 설정(3.3.3) | 위험도 | Medium |
| 대분류 | Storage | | |
| 중분류 | EFS | | |
| 요약 | AWS Backup 백업 계획을 통해 EFS 파일 시스템을 정기적으로 백업하고, 일정 기간 이상 보존하는 것을 권장한다. | | |
| 설명 | AWS Backup은 클라우드 및 온프레미스의 AWS 서비스 전반에서 데이터 백업을 쉽게 중앙집중화 및 자동화할 수 있는 서비스이며, 백업 관리를 간소화하고 비즈니스 및 규제 백업 규정 준수 요구사항을 충족할 수 있다. 백업 계획을 통해 백업 일정, 백업 보존 규칙 및 수명 주기 규칙을 비롯한 백업 요구 사항을 정의하여, 해당 계획에 따라 리소스를 자동으로 백업하고 유지할 수 있다. 기본 백업 기간은 오전 5시(UTC)에 시작하여 8시간 동안 지속된다. 매일(24시간마다) 백업하도록 설정하는 것을 권장한다.   백업 계획을 생성하는 방법은 세 가지가 있다.  - 템플릿으로 시작 : AWS Backup에서 제공하는 템플릿을 기반으로 백업 계획을 생성할 수 있다.   - Daily-35day-Retention (default) : 매일 백업 생성, 35일간 보존  - Daily-Monthly-1yr-Retention : 매일 및 매달 백업 생성, 1년간 보존  - Daily-Weekly-Monthly-5yr-Retention : 매일, 매주, 매달 백업 생성, 5년간 보존  - Daily-Weekly-Monthly-7yr-Retention : 매일, 매주, 매달 백업 생성, 7년간 보존 - 새 계획 수립 : 처음부터 새 백업 계획을 구성할 수 있다.  - JSON을 사용하여 계획 정의 : 기존 백업 계획의 JSON 표현식을 수정하거나 새 표현식을 생성할 수 있다. | | |
| 진단 기준 | ▶ 양호 : AWS Backup 백업 계획에 각 EFS 파일 시스템이 할당되어 정기적으로 백업이 수행되고 있는 경우   ▶ 취약 : AWS Backup 백업 계획에 각 EFS 파일 시스템이 할당되어 있지 않은 경우   ▶ Default Value : - ※ 리소스 자체 백업 기능을 통해 리소스를 백업하고 있는 경우, 해당 항목은 예외처리 할 수 있다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 백업 계획 생성(Create Backup plan) 버튼을 클릭한다.  4. 시작 옵션(Start options)에서 '템플릿으로 시작(Start with a template)'을 선택한다.  5. 템플릿 선택(Choose template)에서 템플릿 계획을 선택한다. 모든 템플릿에는 매일 백업하는 규칙(DailyBackups)이 적용되어 있다.  6. 백업 계획의 이름을 설정한 후, 계획 생성(Create plan) 버튼을 클릭한다.  ※ 시작 옵션에서 '새 계획 수립(Build a new plan)' 또는 'JSON을 사용하여 계획 정의(Define a plan using JSON)'를 선택하여 사용자가 직접 백업 계획을 구성할 수 있으며, 백업 빈도를 "매일(Daily)"로 설정하는 것을 권장한다. 7. 해당 백업 계획으로 이동한다.  8. 리소스 할당(Resource assignments)에서 "리소스 할당(Assign resources)" 버튼을 클릭한다.  9. 리소스 할당 이름 및 IAM 역할을 지정한 후, "다음을 기준으로 할당(Assign by)"에서 "리소스 ID(Resource ID)"를 선택한다.  10. "리소스 유형(Resource type)"에서 "EFS"를 선택하고, "파일 시스템 ID(File system ID)"를 선택한다.  11. 리소스 할당(Assign resources) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 백업 계획을 생성한다.  # aws backup create-backup-plan --backup-plan <value>  예시는 다음과 같다.  # aws backup create-backup-plan --backup-plan "{\"BackupPlanName\":\"Example-Backup-Plan\",\"Rules\":[{\"RuleName\":\"DailyBackups\",\"ScheduleExpression\":\"cron(0 5 ? \* \* \*)\",\"StartWindowMinutes\":480,\"TargetBackupVaultName\":\"Default\",\"Lifecycle\":{\"DeleteAfterDays\":35}}]}"  2. 해당 백업 계획에 리소스를 할당한다.  # aws backup create-backup-selection --backup-plan-id <backup-plan-id> --backup-selection SelectionName="string",IamRoleArn="<iam role arn>",Resources="<resource arn>","<resource arn>",ListOfTags=[{ConditionType=string,ConditionKey=string,ConditionValue=string},{ConditionType=string,ConditionKey=string,ConditionValue=string}] | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 각 백업 계획을 클릭한다.  4. 백업 규칙(Backup rules)이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다.  5. 리소스 할당(Resource assignments)에 EFS 파일 시스템이 지정되어 있는지 확인한다.  ※ 각 EFS 파일 시스템마다 설정된 백업 계획 및 백업 규칙은 다를 수 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 백업 계획의 백업 규칙이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다. # aws backup get-backup-plan --backup-plan-id <backup-plan-id>  2. 백업 계획에 각 EFS 파일 시스템이 할당되어 있는지 확인한다.  # aws backup list-backup-selections --backup-plan-id <backup-plan-id> # aws backup get-backup-selection --backup-plan-id <backup-plan-id> --selection-id <selection id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/efs/latest/ug/awsbackup.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/efs-in-backup-plan.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-efs-2 https://docs.aws.amazon.com/cli/latest/reference/backup/create-backup-plan.html | | |

**Database**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 데이터베이스 퍼블릭 액세스 제한(4.1.1) | 위험도 | High |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | VPC 내부에서만 DB 인스턴스에 접근 가능하도록 퍼블릭 액세스를 차단해야 한다. | | |
| 설명 | RDS 인스턴스의 "퍼블릭 액세스 가능성(publiclyAccessible)" 항목이 "true" 로 설정될 경우, 외부에서 해당 인스턴스에 접근할 수 있으며 악의적인 사용자로부터 데이터가 유출될 수 있으므로 리소스에 대한 액세스를 제어해야 한다. 만약 외부에서 액세스가 필요한 경우 0.0.0.0/0 이 아닌 특정 IP에 한하여 접근 가능하도록 설정해야 하며, 해당 작업이 끝날 경우 즉시 규칙을 삭제하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스에 "퍼블릭 액세스 가능(Public accessibility)" 항목이 "No"로 설정되어 있는 경우   ▶ 취약 : 각 RDS 데이터베이스에 "퍼블릭 액세스 가능" 항목이 "Yes"로 설정되어 있는 경우   ▶ Default Value : No | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스를 클릭한다.  4. 우측 상단의 수정(Modify) 버튼을 클릭한다.  5. '연결(Connectivity)' 항목 하단의 '추가 연결 구성(Additional configuration)'을 클릭한다.  6. "퍼블릭 액세스 가능(Public access)" 항목을 "아니요(Not publicly accessible)"로 설정한다.  7. 계속(Continue) 버튼을 클릭한다.  8. DB 인스턴스 수정(Modify DB instance) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS 에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스를 클릭한다.  4. 연결&보안(Connectivity & security) 탭을 클릭한다.  5. "퍼블릭 액세스 가능(Public accessibility)" 항목이 "아니요(No)"로 설정되어 있는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_VPC.WorkingWithRDSInstanceinaVPC.html https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/Overview.DBInstance.Modifying.html#USER\_ModifyInstance.MySQL.Settings https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_VPC.WorkingWithRDSInstanceinaVPC.html#USER\_VPC.Hiding https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-instance-public-access-check.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-2 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 데이터베이스 암호화 설정(4.1.2) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | Amazon RDS 데이터베이스에 저장 중인 데이터를 암호화하는 것을 권장한다. | | |
| 설명 | Amazon RDS의 암호화된 DB 인스턴스는 AES-256 알고리즘을 사용하여 DB 인스턴스를 호스팅하는 서버의 데이터를 암호화한다. 암호화를 활성화할 경우, 기본 스토리지, 자동 백업, 읽기 복제본 및 스냅샷에 저장된 데이터가 모두 암호화되며 저장된 데이터의 무결성과 기밀성을 보장할 수 있다. 중요한 데이터를 보관하는 데이터베이스의 경우, 무단 액세스로부터 데이터를 보호하기 위해 필수적으로 암호화할 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스의 암호화가 활성화되어 있는 경우   ▶ 취약 : 각 RDS 데이터베이스의 암호화가 비활성화되어 있는 경우   ▶ Default Value : Enabled | | |
| 조치 방안 | ※ Amazon RDS DB 인스턴스에 대한 암호화는 인스턴스를 생성할 때만 설정할 수 있으며, 이미 생성된 이후에는 설정이 불가능하다. 또한, 암호화된 DB 인스턴스의 암호화를 비활성화 할 수 없다. 암호화되지 않은 인스턴스를 암호화하고자 하는 경우, 해당 인스턴스를 삭제한 후 다시 생성해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스를 클릭한다.  4. 구성(Configuration) 탭을 클릭한다.  5. 스토리지(Storage)의 '암호화(Encryption)'가 "활성화(Enabled)"로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 RDS 데이터베이스에 암호화가 설정되어 있는지 확인한다.  # aws rds describe-db-instances --db-instance-identifier <db identifier>  예시는 다음과 같다.  "DBInstances": [  {  "StorageEncrypted": true,  ....  }  ] | | |
| Reference | 2.3.1 Ensure that encryption is enabled for RDS Instances (Automated) | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/Overview.Encryption.html https://aws.amazon.com/ko/rds/features/security/ https://aws.amazon.com/ko/blogs/database/selecting-the-right-encryption-options-for-amazon-rds-and-amazon-aurora-database-engines/#:~:text=With%20RDS%2Dencrypted%20resources%2C%20data,transparent%20to%20your%20database%20engine. https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-storage-encrypted.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-3 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 클러스터 자동 백업 설정(4.1.3) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | RDS 클러스터의 자동 백업을 설정하여 정기적으로 백업하는 것을 권장한다. 백업을 설정할 경우 데이터베이스가 손상되거나 실패할 경우 데이터를 복원하는 데 도움이 된다. | | |
| 설명 | 자동 백업 기능을 활성화할 경우, 특정 시점으로 복구가 가능하므로 데이터를 복원하는데 도움이 된다. 대량의 데이터를 로드하는 등의 특정 상황이 아닐 경우 해당 기능을 활성화하는 것을 권장한다. 만약 특정 상황으로 인해 일시적으로 비활성화해야 하는 경우, 해당 작업이 완료된 이후에 자동 백업을 '즉시 자동 백업을 활성화' 로 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 클러스터의 자동 백업(Automatic backups)이 활성화되어 있고, 백업 보존 기간이 1일 이상으로 설정되어 있는 경우   ▶ 취약 : 각 RDS 클러스터의 자동 백업이 비활성화되어 있거나 0일로 설정되어 있는 경우   ▶ Default Value : 데이터베이스 엔진에 따라 상이함 | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스 클러스터를 클릭한다.  4. 우측 상단의 '수정(Modify)' 버튼을 클릭한다.  5. 추가 구성(Additional configuration)에서 "백업 보존 기간(Backup retention period)" 항목을 1일 이상으로 선택한다.  6. 계속(Continue) 버튼을 클릭한다.  7. 클러스터 수정(Modify cluster) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스 클러스터를 클릭한다.  4. 유지 관리 및 백업(Maintenance & backups) 탭을 클릭한다.  5. 백업(Backup) 의 '자동 백업(Automated backups)' 항목이 "활성화됨(Enabled)" 인지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_WorkingWithAutomatedBackups.html https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_Recommendations.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 스냅샷 퍼블릭 액세스 제한(4.1.6) | 위험도 | High |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | RDS의 DB 스냅샷 가시성은 "Private"으로 설정하는 것을 권장한다. | | |
| 설명 | DB 스냅샷 가시성을 "Public" 으로 설정한 경우, 모든 AWS 계정이 해당 스냅샷에서 DB 인스턴스를 복원하고 사용자 데이터에 액세스 할 수 있게 된다. 민감정보가 저장된 수동 DB 스냅샷의 경우 "Private" 으로 설정하는 것을 권장한다.   - public : 외부에 공개로 설정되어 있는 DB 스냅샷 - automated : 해당 계정 내 RDS에서 자동으로 생성된 모든 DB 스냅샷 - manual : 해당 계정으로 생성한 DB 스냅샷shared : 해당 계정에서 수동으로 생성한 DB 스냅샷 - awsbackup : AWS Backup 서비스로 관리되는 DB 스냅샷 | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스 스냅샷의 "DB 스냅샷 가시성(DB snapshot visibility)" 항목이 "Private"로 설정되어 있는 경우   ▶ 취약 : 각 RDS 데이터베이스 스냅샷의 "DB 스냅샷 가시성" 항목이 "Public"으로 설정되어 있는 경우   ▶ Default Value : - ※ 원본 데이터베이스 클러스터가 암호화되어 있는 경우 private으로 설정되어 있다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '스냅샷(Snapshots)'을 클릭한다.  3. 수동(Manual) 탭을 선택한다.  4. 해당 수동 스냅샷을 클릭한다.  5. 우측 상단의 '작업(Actions)' 메뉴를 클릭하여, '스냅샷 공유(Share snapshot)'를 선택한다.  6. DB 스냅샷 가시성(DB snapshot visibility) 항목을 "Private" 로 설정한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '스냅샷(Snapshots)'을 클릭한다.  3. 수동(Manual) 탭을 선택한다.  4. 각 수동 스냅샷을 클릭한다.  5. 우측 상단의 '작업(Actions)' 메뉴를 클릭하여, '스냅샷 공유(Share snapshot)'를 선택한다.  6. DB 스냅샷 가시성(DB snapshot visibility) 항목이 "Private" 로 설정되어 있는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_ShareSnapshot.html https://docs.aws.amazon.com/AmazonRDS/latest/APIReference/API\_DescribeDBSnapshots.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-snapshots-public-prohibited.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-1 https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_ShareSnapshot.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 클러스터 삭제 방지 활성화(4.1.7) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | 민감정보가 저장된 데이터베이스의 경우, 악의적인 사용자 또는 사용자 실수로 삭제하는 것을 방지하기 위해 삭제 방지 기능을 활성화하는 것을 권장한다. | | |
| 설명 | 삭제 방지(Deletion protection)를 활성화할 경우, 악의적인 사용자 또는 실수로 데이터베이스를 삭제하는 것을 방지하여 데이터에 대한 추가적인 보안 계층을 제공한다. RDS 클러스터를 삭제하고자 할 경우, 삭제 방지 기능을 비활성화한 후 삭제해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 클러스터의 "삭제 방지(Deletion protection)" 항목이 활성화되어 있는 경우   ▶ 취약 : 각 RDS 클러스터의 "삭제 방지" 항목이 비활성화되어 있는 경우   ▶ Default Value : 데이터베이스 엔진에 따라 상이함 | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스를 클릭한다.  4. 우측 상단의 수정(Modify) 버튼을 클릭한다.  5. "삭제 방지 활성화(Enable deletion protection)" 항목을 체크한다.  6. 계속(Continue) 버튼을 클릭한다.  7. 클러스터 수정(Modify cluster) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 RDS 클러스터의 "DeletionProtection" 항목을 "true"로 설정한다.  # aws rds modify-db-cluster --db-cluster-identifier <database identifier> --deletion-protection | | |
| 감사 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스를 클릭한다.  4. 구성(Configuration) 탭을 클릭한다.  5. '삭제 방지(Deletion protection)' 항목이 "활성화됨(Enabled)" 인지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 각 RDS 클러스터의 "DeletionProtection" 항목이 "true"로 설정되어 있는지 확인한다.  # aws rds describe-db-instances | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_DeleteInstance.html#USER\_DeleteInstance.DeletionProtection https://docs.aws.amazon.com/cli/latest/reference/rds/modify-db-cluster.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-cluster-deletion-protection-enabled.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-7 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 백업 계획 설정(4.1.8) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | AWS Backup 백업 계획을 통해 RDS 데이터베이스를 정기적으로 백업하고, 일정 기간 이상 보존하는 것을 권장한다. | | |
| 설명 | AWS Backup은 클라우드 및 온프레미스의 AWS 서비스 전반에서 데이터 백업을 쉽게 중앙집중화 및 자동화할 수 있는 서비스이며, 백업 관리를 간소화하고 비즈니스 및 규제 백업 규정 준수 요구사항을 충족할 수 있다. 백업 계획을 통해 백업 일정, 백업 보존 규칙 및 수명 주기 규칙을 비롯한 백업 요구 사항을 정의하여, 해당 계획에 따라 리소스를 자동으로 백업하고 유지할 수 있다. 기본 백업 기간은 오전 5시(UTC)에 시작하여 8시간 동안 지속된다. 매일(24시간마다) 백업하도록 설정하는 것을 권장한다.   백업 계획을 생성하는 방법은 세 가지가 있다.  - 템플릿으로 시작 : AWS Backup에서 제공하는 템플릿을 기반으로 백업 계획을 생성할 수 있다.   - Daily-35day-Retention (default) : 매일 백업 생성, 35일간 보존  - Daily-Monthly-1yr-Retention : 매일 및 매달 백업 생성, 1년간 보존  - Daily-Weekly-Monthly-5yr-Retention : 매일, 매주, 매달 백업 생성, 5년간 보존  - Daily-Weekly-Monthly-7yr-Retention : 매일, 매주, 매달 백업 생성, 7년간 보존 - 새 계획 수립 : 처음부터 새 백업 계획을 구성할 수 있다.  - JSON을 사용하여 계획 정의 : 기존 백업 계획의 JSON 표현식을 수정하거나 새 표현식을 생성할 수 있다. | | |
| 진단 기준 | ▶ 양호 : AWS Backup 백업 계획에 각 RDS 데이터베이스가 할당되어 정기적으로 백업이 수행되고 있는 경우   ▶ 취약 : AWS Backup 백업 계획에 각 RDS 데이터베이스가 할당되어 있지 않은 경우   ▶ Default Value : - ※ 리소스 자체 백업 기능을 통해 리소스를 백업하고 있는 경우, 해당 항목은 예외처리 할 수 있다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 백업 계획 생성(Create Backup plan) 버튼을 클릭한다.  4. 시작 옵션(Start options)에서 '템플릿으로 시작(Start with a template)'을 선택한다.  5. 템플릿 선택(Choose template)에서 템플릿 계획을 선택한다. 모든 템플릿에는 매일 백업하는 규칙(DailyBackups)이 적용되어 있다.  6. 백업 계획의 이름을 설정한 후, 계획 생성(Create plan) 버튼을 클릭한다.  ※ 시작 옵션에서 '새 계획 수립(Build a new plan)' 또는 'JSON을 사용하여 계획 정의(Define a plan using JSON)'를 선택하여 사용자가 직접 백업 계획을 구성할 수 있으며, 백업 빈도를 "매일(Daily)"로 설정하는 것을 권장한다. 7. 해당 백업 계획으로 이동한다.  8. 리소스 할당(Resource assignments)에서 "리소스 할당(Assign resources)" 버튼을 클릭한다.  9. 리소스 할당 이름 및 IAM 역할을 지정한 후, "다음을 기준으로 할당(Assign by)"에서 "리소스 ID(Resource ID)"를 선택한다.  10. "리소스 유형(Resource type)"에서 "RDS"를 선택하고, "데이터베이스 이름(Database name)"을 선택한다.  11. 리소스 할당(Assign resources) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 백업 계획을 생성한다.  # aws backup create-backup-plan --backup-plan <value>  예시는 다음과 같다.  # aws backup create-backup-plan --backup-plan "{\"BackupPlanName\":\"Example-Backup-Plan\",\"Rules\":[{\"RuleName\":\"DailyBackups\",\"ScheduleExpression\":\"cron(0 5 ? \* \* \*)\",\"StartWindowMinutes\":480,\"TargetBackupVaultName\":\"Default\",\"Lifecycle\":{\"DeleteAfterDays\":35}}]}"  2. 해당 백업 계획에 리소스를 할당한다.  # aws backup create-backup-selection --backup-plan-id <backup-plan-id> --backup-selection SelectionName="string",IamRoleArn="<iam role arn>",Resources="<resource arn>","<resource arn>",ListOfTags=[{ConditionType=string,ConditionKey=string,ConditionValue=string},{ConditionType=string,ConditionKey=string,ConditionValue=string}] | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 각 백업 계획을 클릭한다.  4. 백업 규칙(Backup rules)이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다.  5. 리소스 할당(Resource assignments)에 RDS 데이터베이스가 지정되어 있는지 확인한다.  ※ 각 RDS 데이터베이스마다 설정된 백업 계획 및 백업 규칙은 다를 수 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 백업 계획의 백업 규칙이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다. # aws backup get-backup-plan --backup-plan-id <backup-plan-id>  2. 백업 계획에 각 RDS 데이터베이스가 할당되어 있는지 확인한다.  # aws backup list-backup-selections --backup-plan-id <backup-plan-id> # aws backup get-backup-selection --backup-plan-id <backup-plan-id> --selection-id <selection id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER\_WorkingWithAutomatedBackups.html#AutomatedBackups.AWSBackup https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER\_Tagging.html#Tagging.RDS.AWSBackup https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-in-backup-plan.html https://docs.aws.amazon.com/cli/latest/reference/backup/create-backup-plan.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 로깅 활성화(4.1.11) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | RDS 데이터베이스의 로그를 CloudWatch Logs로 게시하는 것을 권장한다. | | |
| 설명 | RDS 로깅을 활성화할 경우, RDS 데이터베이스에 액세스한 사용자의 정보, IP 등 요청의 세부 정보를 확인할 수 있으며, 보안, 액세스 감사, 가용성 문제 진단 등에 도움이 될 수 있다. 로깅을 활성화하여 Amazon CloudWatch Logs로 내보내는 것을 권장한다. CloudWatch Logs를 통해 로그 데이터에 대한 실시간 분석을 수행하거나 경보를 생성하여 지표를 확인할 수 있다. 또한, 지표가 지정된 임계값을 초과할 경우 사용자에게 알림을 전송할 수 있다. CloudWatch Logs에 로그 내보내기를 활성화할 경우, 각 데이터베이스 엔진의 로그 유형에 대해 로그 그룹이 생성되며, 보존 기간을 지정하지 않는 한 로그를 무기한 보존할 수 있다. 침해사고 조사 및 대응을 위해 최소 1년 이상 로그 데이터를 보존하는 것을 권장한다.   각 데이터베이스 엔진에서 CloudWatch Logs로 내보내기가 가능한 로그 유형은 다음과 같다.  - Amazon Aurora   - MySQL과 호환되는 Aurora : 감사 로그, 에러 로그, 일반 로그, 느린 쿼리 로그  - PostgreSQL과 호환되는 Aurora : PostgreSQL 로그 - MySQL : 감사 로그, 에러 로그, 일반 로그, 느린 쿼리 로그 - MariaDB : 감사 로그, 에러 로그, 일반 로그, 느린 쿼리 로그 - PostgreSQL : Postgresql 로그, 업그레이드 로그 - Oracle : 알림 로그, 감사 로그, 리스너 로그, Oracle Management Agent log, 추적 로그 - Microsoft SQL : 에러 로그, 에이전트 로그 | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스의 로그를 CloudWatch Logs에 게시하는 경우   ▶ 취약 : 각 RDS 데이터베이스의 로그를 CloudWatch Logs에 게시하지 않는 경우  ▶ Default Value : - | | |
| 조치 방안 | ※ 새로운 RDS 데이터베이스를 생성하는 경우  [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 데이터베이스 생성(Create database) 버튼을 클릭한다.  4. 엔진 옵션, 템플릿, 인스턴스 크기 등을 설정한 후, '추가 구성(Additional configuration)'을 클릭하여 "로그 내보내기(Log exports)" 항목의 로그 유형을 체크한다.  5. 데이터베이스 생성(Create database) 버튼을 클릭한다.   ※ 기존 RDS 데이터베이스에서 수정하는 경우 [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스를 클릭한다.  4. 수정(Modify) 버튼을 클릭한다.  5. 추가 구성(Additional configuration)의 "로그 내보내기(Log exports)" 항목의 로그 유형을 체크한다.  6. 하단의 '계속(Continue)' 버튼을 클릭한다.  7. 수정 사항을 적용할 시간을 선택한 후, 'DB 인스턴스 수정(Modify DB instance)' 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 RDS 데이터베이스의 로그를 CloudWatch Logs에 게시한다.  # aws rds modify-db-instance --db-instance-identifier <db identifier> --cloudwatch-logs-export-configuration EnableLogTypes=<value> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스를 클릭한다.  4. 구성(Configuration) 탭을 클릭한다.  5. 데이터베이스(Database) 의 "게시된 로그(Published logs)" 항목에 로그 유형이 설정되어 있는지 확인한다.  ※ RDS 데이터베이스 엔진에 따라 CloudWatch Logs로 게시할 수 있는 로그 유형은 상이할 수 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 RDS 데이터베이스의 로그를 CloudWatch Logs에 게시하고 있는지 확인한다.  # aws rds describe-db-instances --db-instance-identifier <db identifier>   예시는 다음과 같다.  "DBInstances": [  {  "EnabledCloudwatchLogsExports": [  "audit",  "error" ]  }  ] | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.LoggingAndMonitoring.html https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER\_LogAccess.html https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/publishing\_cloudwatchlogs.html https://docs.aws.amazon.com/config/latest/developerguide/rds-logging-enabled.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-9 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 데이터베이스 확장 모니터링 활성화(4.1.12) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | RDS 데이터베이스에 확장 모니터링을 설정하는 것을 권장한다. | | |
| 설명 | 확장 모니터링을 활성화할 경우, DB 인스턴스가 실행되는 운영체제에 대한 측정치를 실시간으로 확인할 수 있으며 여러 프로세스 또는 스레드에서 CPU를 사용하는 방법을 확인하려는 경우에 유용하다. CloudWatch를 사용하여 RDS 데이터베이스를 모니터링 할 경우, DB 인스턴스의 하이퍼바이저에서 CPU 사용률에 대한 측정치만을 수집하며 소량의 작업만 수행한다. 반면에, 확장 모니터링의 경우에는 인스턴스의 에이전트에서 측정치를 수집한다. 또한, 모니터랑 간격을 지정하여 사용자가 원하는 간격마다 측정할 수 있으며, 확장 모니터링 지표는 CloudWatch Logs에 30일간 저장된다. CloudWatch Logs로 로그를 보내기 위해서는 IAM 역할을 사용하여 확장 모니터링에 권한을 부여해주어야 한다.   ※ 확장 모니터링을 사용할 경우 추가적인 비용이 발생할 수 있다. 확장 모니터링 기능은 db.m1.small 인스턴스 클래스를 제외한 모든 DB 인스턴스 클래스에서 사용할 수 있으며, 사용 가능한 데이터베이스 엔진은 다음과 같다.  - MariaDB - Microsoft SQL Server - MySQL 버전 5.5 이상 - Oracle - PostgreSQL | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스의 확장 모니터링(Enhanced monitoring)이 활성화되어 있는 경우   ▶ 취약 : 각 RDS 데이터베이스의 확장 모니터링이 비활성화되어 있는 경우  ▶ Default Value : 데이터베이스 엔진에 따라 다름 | | |
| 조치 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스 인스턴스를 클릭한다. ­­­ 4. 우측 상단의 '수정(Edit)' 버튼을 클릭한다.  5. 추가 구성(Additional configuration) 탭의 "Enhanced 모니터링 활성화(Enable Enhanced monitoring)" 항목을 체크한다. 6. 세부 수준(Granularity)과 역할 모니터링(Monitoring Role)을 설정한다.  7. 하단의 '계속(Continue)' 버튼을 클릭한다.  8. 수정 사항을 적용할 시간을 선택한 후, 'DB 인스턴스 수정(Modify DB instance)' 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 RDS 데이터베이스의 확장 모니터링을 활성화한다. "monitoring-interval"의 값을 1 이상으로 설정해야 한다.  # aws rds modify-db-instance --db-instance-identifier <db identifier> --monitoring-interval=<value> ※ Valid Values : 0, 1, 5, 10, 15, 30, 60 | | |
| 감사 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스 인스턴스를 클릭한다.  4. 우측 상단의 '수정(Edit)' 버튼을 클릭한다.  5. 추가 구성(Additional configuration) 탭의 "Enhanced 모니터링 활성화(Enable Enhanced monitoring)" 항목이 체크되어 있는지 확인한다.   [AWS CLI]  1. 다음 명령어를 실행하여 RDS 데이터베이스의 확장 모니터링이 활성화되어 있는지 확인한다. "MonitoringInterval" 항목이 1 이상으로 설정되어 있어야 한다. 0으로 설정된 경우, 확장 모니터링이 비활성화됨을 의미한다.  # aws rds describe-db-instances --db-instance-identifier <db identifier>  예시는 다음과 같다.  "DBInstances": [  {  "MonitoringInterval": 60,  "MonitoringRoleArn": "<role arn>",   ....  }  ] | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER\_Monitoring.OS.html https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_Recommendations.html https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/CHAP\_BestPractices.html#CHAP\_BestPractices.EnhancedMonitoring https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-enhanced-monitoring-enabled.html https://docs.aws.amazon.com/ko\_kr/AmazonRDS/latest/UserGuide/USER\_Monitoring.OS.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-6 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 스냅샷 암호화(4.1.13) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | RDS 데이터베이스의 스냅샷을 암호화하는 것을 권장한다. | | |
| 설명 | 데이터베이스의 스냅샷에 암호화를 활성화할 경우, 스냅샷에 저장된 데이터의 무결성과 기밀성을 보장할 수 있다. 데이터베이스를 암호화할 경우 자동적으로 스냅샷도 암호화된다. 중요한 데이터를 보관하는 데이터베이스의 경우, 무단 액세스로부터 데이터를 보호하기 위해 암호화를 필수적으로 설정하는 것을 권장한다.  ※ Amazon RDS DB 인스턴스에 대한 암호화는 인스턴스를 생성할 때만 설정할 수 있으며, 이미 생성된 이후에는 설정이 불가능하다. 또한, 암호화된 DB 인스턴스의 암호화를 비활성화 할 수 없다. 암호화되지 않은 인스턴스를 암호화하고자 하는 경우, 해당 인스턴스를 삭제한 후 다시 생성해야 한다.  ※ 암호화되지 않은 DB 인스턴스의 암호화된 스냅샷은 생성할 수 없다.  ※ 암호화된 DB 인스턴스의 스냅샷은 DB 인스턴스와 동일한 CMK를 사용하여 암호화해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스의 스냅샷이 암호화되어 있는 경우  ▶ 취약 : 각 RDS 데이터베이스의 스냅샷이 암호화되어 있지 않은 경우  ▶ Default Value : 데이터베이스의 암호화가 된 경우 스냅샷도 암호화되지만, 데이터베이스가 암호화되지 않은 경우 스냅샷도 암호화되지 않음 | | |
| 조치 방안 | ※ Amazon RDS 스냅샷에 암호화를 설정하기 위해서는 데이터베이스 인스턴스에 대한 암호화가 설정되어 있어야 한다. 그러나 데이터베이스 암호화는 인스턴스를 생성할 때만 설정할 수 있으며 이미 생성된 이후에는 설정이 불가능하다. 암호화되지 않은 인스턴스를 암호화하고자 하는 경우, 해당 인스턴스를 삭제한 후 다시 생성해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다. 2. 좌측 사이드 바에서 '스냅샷(Snapshots)'을 클릭한다.  3. 수동(Manual) 탭을 선택한다.  4. 각 수동 스냅샷을 클릭한다.  5. 세부 정보(Details)의 KMS 키 ID(KMS key ID)가 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 RDS 데이터베이스 스냅샷이 암호화되어 있는지 확인한다.  # aws rds describe-db-snapshots --db-instance-identifier <db identifier>  예시는 다음과 같다.   "DBSnapshots": [  {  ...  "Encrypted": true,  ...  }  ] | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-snapshot-encrypted.html https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-4 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 데이터베이스 다중 AZ 배포 설정(4.1.14) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | RDS 데이터베이스에 다중 AZ 배포를 설정하는 것을 권장한다. | | |
| 설명 | Amazon RDS는 다중 AZ 지원을 통해 데이터베이스 인스턴스의 가용성과 내구성을 강화할 수있다. 다중 AZ 지원을 활성화할 경우, RDS는 자동으로 서로 다른 가용 영역에 동기식 예비 복제본을 프로비저닝하고 유지한다. 기본 DB 인스턴스는 가용 영역에서 예비 복제본으로 동기식으로 복제되어, 데이터 이중화를 제공하고 I/O 중지를 제거하며 시스템 백업 시 지연 시간을 최소화한다. 또한 DB 인스턴스에 계획되거나 계획되지 않은 중단이 발생할 경우, 자동으로 다른 가용 영역에 있는 예비 복제본으로 전환하여 장애를 조치할 수 있다. 각 가용 영역은 물리적으로 구분된 독립 인프라를 기반으로 실행되며 높은 안정성을 제공하므로 다중 AZ 지원 기능을 활성화하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 데이터베이스의 "다중 AZ(Multi AZ)"가 활성화되어 있는 경우   ▶ 취약 : 각 RDS 데이터베이스의 "다중 AZ"가 비활성화되어 있는 경우  ▶ Default Value : 데이터베이스 엔진에 따라 다름 | | |
| 조치 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스 인스턴스를 클릭한다.  4. 상단의 '수정(Modify)' 버튼을 클릭한다.  5. 가용성 및 내구성(Availability & durability)의 "다중 AZ 배포(Multi-AZ deployment)" 항목에서 "대기 인스턴스 생성(생산 사용량에 권장)(Create a standby instance (recommended for production usage))"를 체크한다.  6. 계속(Continue) 버튼을 클릭한다.  7. 수정 사항을 적용할 시간을 선택한 후, 'DB 인스턴스 수정(Modify DB instance)' 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 다중 AZ를 활성화한다.  # aws rds modify-db-instance --db-instance-identifier <db identifier> --multi-az | | |
| 감사 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스 인스턴스를 클릭한다.  4. 구성(Configuration) 탭을 클릭한다.  5. 다중 AZ(Multi-AZ) 항목이 "예(Yes)"로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 다중 AZ가 활성화되어 있는지 확인한다.  # aws rds describe-db-instances --db-instance-identifier <db identifier>  예시는 다음과 같다.  "DBInstances": [  {  ....  "MultiAZ": true,  ....  }  ] | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER\_UpgradeDBInstance.Maintenance.html#USER\_UpgradeDBInstance.Maintenance.Multi-AZ https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.MultiAZ.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/rds-multi-az-support.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-rds-5 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | RDS 리소스 태그 지정(4.1.15) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | RDS | | |
| 요약 | 각 RDS 리소스에 태그를 지정하는 것을 권장한다. | | |
| 설명 | Amazon RDS 태그는 사용자가 정의한 Amazon RDS 리소스와 연결하는 이름-값 쌍이며, 각 리소스에 최대 50개의 태그를 지정할 수 있다. 태그를 설정할 경우, 데이터베이스, 인스턴스, 스냅샷, 클러스터에 따로 고유하게 표기를 할 수 있어 RDS 리소스를 문서화하거나 자동화된 유지 관리 절차에 도움이 될 수 있다. 또한, IAM 정책과 태그를 함께 사용하여 RDS 리소스에 대한 액세스를 관리하고 적용 가능한 작업을 제어할 수 있다. 비슷하게 태그가 지정된 리소스에 대한 비용을 그룹화하여 비용을 추적할 수도 있다.  ※ 사용자 지정 태그 식별자는 대/소문자를 구분한다. | | |
| 진단 기준 | ▶ 양호 : 각 RDS 리소스에 태그가 지정되어 있는 경우   ▶ 취약 : 각 RDS 리소스에 태그가 지정되어 있지 않은 경우  ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 해당 데이터베이스 인스턴스를 클릭한다.  4. 태그(Tags) 탭을 클릭한다.  5. 추가(Add) 버튼을 클릭한다.  6. "태그 키(Tag key)"와 "값(Value)"을 입력한 후, 추가(Add) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 RDS 리소스에 태그를 지정한다.  # aws rds add-tags-to-resource --resource-name <db instance arn> --tags <value> Key=<tag key>,Value=<tag value> | | |
| 감사 방안 | [AWS Console]  1. https://console.aws.amazon.com/rds 를 통해 Amazon RDS에 접속한다.  2. 좌측 사이드 바에서 '데이터베이스(Databases)'를 클릭한다.  3. 각 데이터베이스 인스턴스를 클릭한다.  4. 태그(Tags) 탭을 클릭한다.  5. 리소스에 지정된 태그가 존재하는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 RDS 리소스에 태그가 지정되어 있는지 확인한다.  # aws rds list-tags-for-resource --resource-name <db instance arn> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER\_Tagging.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Redshift 클러스터 공개적 액세스 설정(4.2.1) | 위험도 | High |
| 대분류 | Database | | |
| 중분류 | Redshift | | |
| 요약 | Redshift 클러스터 네트워크 보안 설정 시 공개적으로 액세스하지 못하도록 설정해야 한다. | | |
| 설명 | 운영상 필요한 경우가 아닐 경우 Redshift 클러스터는 퍼블릭 엔드포인트를 설정하지 않아야 하며, VPC 내에서만 접근 가능하도록 설정해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 Redshift 클러스터의 "공개적으로 액세스할 수 있음(Publicly accessible)" 항목이 비활성화되어 있는 경우   ▶ 취약 : 각 Redshift 클러스터의 "공개적으로 액세스할 수 있음" 항목이 활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 해당 클러스터를 클릭한다.  4. 속성(Properties) 탭을 클릭한다.  5. Network and security settings 에서 'Edit publicly accessible' 버튼을 클릭한다.  6. 공개적으로 액세스할 수 있음(Publicly accessible) 항목을 "비활성화(Disable)"로 설정한다.  7. Save changes 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 "PubliclyAccessible" 항목을 "false"로 설정한다.  # aws redshift modify-cluster --cluster-identifier <cluster identifier> --no-publicly-accessible | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 속성(Properties) 탭을 클릭한다.  5. Network and security settings 에서 '공개적으로 액세스할 수 있음(Publicly accessible)' 항목이 "비활성화(Disabled)" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 "PubliclyAccessible" 항목이 "false"로 설정되어 있는지 확인한다.  # aws redshift describe-clusters | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://aws.amazon.com/ko/premiumsupport/knowledge-center/private-redshift-cluster-local-machine/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Redshift 클러스터 암호화 설정(4.2.2) | 위험도 | High |
| 대분류 | Database | | |
| 중분류 | Redshift | | |
| 요약 | Redshift 클러스터에 저장 중인 데이터를 암호화하는 것을 권장한다. 클러스터에서 암호화를 활성화할 경우 해당 클러스터와 스냅샷의 데이터 블록 및 시스템 메타데이터가 암호화된다. | | |
| 설명 | Amazon Redshift 클러스터에 데이터 보관 시 암호화를 활성화할 경우 저장된 데이터의 무결성과 기밀성을 보장할 수 있다. 개인정보와 같은 민감정보를 보관하고 있는 경우 암호화를 필수적으로 설정해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 Redshift 클러스터의 암호화가 활성화되어 있는 경우   ▶ 취약 : 각 Redshift 클러스터의 암호화가 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 해당 클러스터를 클릭한다.  4. 데이터베이스 구성(Database configurations)의 '편집(Edit)' 버튼을 클릭하여, 'Edit encryption' 을 선택한다.  5. 암호화(Encryption) 항목에서 "AWS Key Management Service(AWS KMS) 사용(Use AWS Key Management Service (AWS KMS))" 또는 "HSM(하드웨어 보안 모듈) 사용(Use a hardware security module (HSM))"을 선택한다.  6. Save changes 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 "Encrypted" 항목을 "true"로 설정한다.  # aws redshift modify-cluster --cluster-identifier <cluster identifier> --encrypted --kms-key-id <kms key id> ※ KmsKeyId 값을 입력할 경우 해당 kms key로 암호화되며, 입력하지 않을 경우 기본 키로 암호화된다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 속성(Properties) 탭을 클릭한다.  5. 데이터베이스 구성(Database configurations)의 '암호화(Encryption)' 항목이 "활성화(Enabeld)" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 "Encrypted" 항목이 "true"로 설정되어 있는지 확인한다.  # aws redshift describe-clusters | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/redshift/latest/mgmt/working-with-db-encryption.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Redshift 클러스터 로깅 활성화(4.2.3) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | Redshift | | |
| 요약 | Redshift 클러스터의 감사 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | Amazon Redshift의 감사 로깅을 활성화할 경우, 연결 로그, 사용자 로그, 사용자 작업 로그를 S3 버킷에 기록한다. 연결 로그는 요청한 사용자의 IP 주소, 요청 시기, 사용한 인증 유형, 인증 시도 횟수와 연결 및 연결 해제를 기록하며, 데이터베이스에 연결 중인 사용자에 대한 정보 및 관련 연결 정보를 모니터링 할 수 있다. 사용자 로그는 데이터베이스 사용자 정의 변경 사항에 대한 정보를 기록하며, 사용자 작업 로그는 사용자와 시스템이 데이터베이스에서 실행하는 쿼리 유형에 대한 정보를 기록한다. 보안 문제 해결, 액세스 감사, 가용성 문제 진단 등을 위해 감사 로깅을 활성화하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 Redshift 클러스터의 감사 로깅(Audit logging)이 활성화되어 있는 경우   ▶ 취약 : 각 Redshift 클러스터의 감사 로깅이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 해당 클러스터를 클릭한다.  4. 데이터베이스 구성(Database configurations)의 '편집(Edit)' 버튼을 클릭하여, 'Edit audit logging' 을 선택한다.  5. 감사 로깅 활성화(Configure audit logging) 항목에서 "활성화(Enable)"를 선택한 후, 로그를 저장할 S3 버킷을 지정한다.  6. Save changes 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 로깅을 활성화한다.  # aws redshift enable-logging --cluster-identifier <cluster identifier> --bucket-name <s3 bucket name> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 속성(Properties) 탭을 클릭한다.  5. 데이터베이스 구성(Database configurations)의 '감사 로깅(Audit logging)' 항목이 "활성화(Enabeld)" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 로깅이 활성화되어 있는지 확인한다.  # aws redshift describe-logging-status --cluster-identifier <cluster identifier> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/redshift/latest/mgmt/db-auditing.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Redshift 클러스터 버전 업그레이드 설정(4.2.4) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | Redshift | | |
| 요약 | Redshift 클러스터에 주기적인 유지 관리와 버전 업그레이드를 허용하도록 설정하는 것을 권장한다. | | |
| 설명 | Amazon Redshift는 정기적으로 유지 관리를 수행하여 클러스터를 업그레이드한다. 유지 관리 기간을 설정하거나 필수가 아닌 업데이트 항목을 연기할 수 있으며, 유지 관리 트랙을 통해 클러스터가 업그레이드되는 버전을 선택할 수 있다. 버전 업그레이드 허용을 설정할 경우, 유지 관리 기간 동안 Amazon Redshift 클러스터에 대한 최신 버전으로 업데이트 된다. 최신 버전에는 보안 패치 및 버그 수정 등이 포함될 수 있으며, 보안을 위해 버전을 업그레이드하는 것을 권장한다.  ※ 기본 유지 관리 기간은 리전마다 상이하다. | | |
| 진단 기준 | ▶ 양호 : 각 Redshift 클러스터의 "버전 업그레이드 허용(Allow version upgrade)" 항목이 설정되어 있는 경우   ▶ 취약 : 각 Redshift 클러스터의 "버전 업그레이드 허용" 항목이 설정되어 있지 않은 경우   ▶ Default Value : Enabled | | |
| 조치 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 자동 버전 업그레이드를 활성화한다.  # aws redshift modify-cluster --cluster-identifier <cluster identifier> --allow-version-upgrade | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 'Maintenance' 탭을 클릭한다.  5. 'Maintenance detail' 의 버전 업그레이드 허용(Allow version upgrade)' 항목이 "예(Yes)"로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 "AllowVersionUpgrade" 항목이 "true"로 설정되어 있는지 확인한다.  # aws redshift describe-clusters --cluster-identifier <cluster identifier> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/redshift/latest/mgmt/working-with-clusters.html#rs-cluster-maintenance https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/redshift-cluster-maintenancesettings-check.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-redshift-6 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Redshift 클러스터 전송 중 암호화 설정(4.2.5) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | Redshift | | |
| 요약 | Redshift 클러스터에 전송 중 암호화를 설정하는 것을 권장한다. | | |
| 설명 | Amazon Redshift는 데이터와 서버 인증서 암호화를 위한 SSL(Secure Sockets Layer) 연결을 지원하여 클라이언트에서 연결할 서버 인증서를 검증한다. Redshift 클러스터는 기본적으로 SSL의 사용 여부와 상관없이 연결을 허용한다. 악의적인 공격자가 네트워크 트래픽을 도청하거나 조작하는 것을 방지하기 위해, 클러스터의 연결에 전송 중 암호화를 설정하는 것을 권장한다.  ※ 클러스터가 SSL 연결을 요구하도록 구성하려면 클러스터와 연결된 파라미터 그룹에서 "require\_SSL" 파라미터를 "true" 로 설정해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 Redshift 클러스터에 연결할 때 SSL(Secure Sockets Layer)이 설정되어 있는 경우   ▶ 취약 : 각 Redshift 클러스터에 연결할 때 SSL이 설정되어 있지 않은 경우   ▶ Default Value : Not set | | |
| 조치 방안 | JDBC(Java Database Connectivity) 및 ODBC(Open Database Connectivity) 연결을 통해 SQL 클라이언트 도구에서 Redshift 클러스터에 연결할 수 있다. 아래 URL을 참고하여 클러스터에 연결할 때 SSL을 설정하는 것을 권장한다.  ※ https://docs.aws.amazon.com/ko\_kr/redshift/latest/mgmt/connecting-ssl-support.html ※ https://docs.aws.amazon.com/ko\_kr/redshift/latest/mgmt/configuring-connections.html | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/redshiftv2/ 를 통해 Amazon Redshift에 접속한다. 2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 속성(Properties) 탭을 클릭한다. 5. 데이터베이스 구성(Database configurations)에서 "파라미터 그룹(Parameter group)"을 클릭한다.  6. 파라미터(Parameters) 탭을 선택한다.  7. "require\_ssl" 파라미터가 "true"로 설정되어 있는지 확인한다.  ※ 파라미터 그룹이 Default parameter group으로 설정되어 있는 경우, "require\_ssl" 파라미터는 "false"로 설정되어 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 Redshift 클러스터의 파라미터 그룹에 "require\_ssl" 파라미터가 "true"로 설정되어 있는지 확인한다.  # aws redshift describe-cluster-parameters --parameter-group-name <parameter group name>  예시는 다음과 같다.  {  "ParameterName": "require\_ssl",  "ParameterValue": "true",  "Description": "require ssl for all databaseconnections",  ... } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/redshift/latest/mgmt/security-encryption-in-transit.html https://docs.aws.amazon.com/redshift/latest/mgmt/connecting-ssl-support.html https://docs.aws.amazon.com/config/latest/developerguide/redshift-require-tls-ssl.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-redshift-2 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | DynamoDB 암호화 설정(4.3.1) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | DynamoDB | | |
| 요약 | DynamoDB 테이블 내 데이터를 KMS 키로 암호화하는 것을 권장한다. | | |
| 설명 | DynamoDB에 저장된 모든 데이터는 기본적으로 AWS 소유 CMK를 통해 암호화된다. DynamoDB 유휴 데이터 암호화는 AWS KMS(Key Management Service)와 통합하여 테이블을 암호화하는데 사용되는 키를 관리한다. 민감정보가 담긴 테이블의 경우, KMS의 고객 관리형 키(Customer-managed key)를 통해 암호화는 것을 권장한다. 고객 관리형 키로 암호화할 경우, AWS 관리형 키(AWS-managed key)보다 암호화 및 복호화 프로세스를 보다 세부적으로 제어할 수 있다.  유휴 데이터 암호화에 사용되는 고객 마스터 키(Customer master key)는 다음과 같다.  - AWS 소유 CMK(기본값) : 키는 DynamoDB에서 직접 소유한다. 해당 CMK 사용 시 추가적인 요금이 부과되지 않는다.  - KMS - 고객 관리형 CMK : 키는 사용자가 생성, 소유, 관리하는 계정에 저장된다. 해당 CMK 사용 시 KMS 비용이 부과된다.  - KMS - AWS 관리형 CMK : 키는 사용자 계정에 저장되며, KMS에서 관리한다. 해당 CMK 사용 시 KMS 비용이 부과된다. | | |
| 진단 기준 | ▶ 양호 : 각 DynamoDB 테이블이 "KMS - AWS 관리형 키 CMK(KMS - AWS managed CMK)" 또는 "KMS - 고객 관리형 키 CMK(KMS - Customer managed CMK)"로 암호화되어 있는 경우   ▶ 취약 : 각 DynamoDB 테이블이 암호화되어 있지 않거나 "기본값(Default)"으로 암호화되어 있는 경우   ▶ Default Value : DynamoDB 테이블 생성 시 '기본 설정(Default setting)' 을 사용할 경우, 기본 암호화 유형(Default)으로 암호화됨 | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/dynamodb/ 를 통해 AWS DynamoDB에 접속한다.  2. 좌측 사이드 바에서 '테이블(Table)'을 클릭한다. 3. 해당 테이블을 클릭한다.  4. 테이블 세부 정보(Table details) 에서 '암호화 유형(Encryption Type)' 항목의 "암호화 관리(Manage Encryption)"를 클릭한다.  5. "KMS - AWS 관리형 키 CMK(KMS - AWS managed CMK)" 또는 "KMS - 고객 관리형 CMK(KMS - Customer managed CMK)"를 선택한 후, 저장(Save)을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/dynamodb/ 를 통해 AWS DynamoDB에 접속한다.  2. 좌측 사이드 바에서 '테이블(Table)'을 클릭한다. 3. 각 테이블을 클릭한다.  4. 테이블 세부 정보(Table details)에서 '암호화 유형(Encryption Type)' 항목이 "KMS"로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 DynamoDB 테이블이 KMS 키로 암호화되어 있는지 확인한다.  # aws dynamodb describe-table --table-name <table name>  예시는 다음과 같다.  "SSEDescription": {  "Status": "ENABLED",  "SSEType": "KMS",  "KMSMasterKeyArn": "key arn" } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/EncryptionAtRest.html  https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/best-practices-security-preventative.html https://docs.aws.amazon.com/config/latest/developerguide/dynamodb-table-encrypted-kms.html  https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | DynamoDB 특정 시점으로 복구 활성화(4.3.2) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | DynamoDB | | |
| 요약 | DynamoDB 테이블의 특정 시점 복구를 활성화하는 것을 권장한다. | | |
| 설명 | 특정 시점 복구(PITR, Point-in-Time Recovery)는 DynamoDB 테이블의 백업을 자동화하며, 실수로 쓰기 또는 삭제 작업이 발생했을 때 복구하는데 걸리는 시간을 줄일 수 있다. 특정 시점 복구 기능이 활성화된 경우, 지난 35일 이내의 원하는 시점으로 복원이 가능하다. | | |
| 진단 기준 | ▶ 양호 : 각 DynamoDB 테이블의 "특정 시점으로 복구(Point-in-time recovery)"가 활성화되어 있는 경우   ▶ 취약 : 각 DynamoDB 테이블의 "특정 시점으로 복구"가 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/dynamodb/ 를 통해 AWS DynamoDB에 접속한다.  2. 좌측 사이드 바에서 '테이블(Table)'을 클릭한다. 3. 해당 테이블을 클릭한다.  4. 개요(Overview) 탭을 선택한다.  5. 테이블 세부 정보(Table details)에서 '특정 시점으로 복구(Point-in-time recovery)' 항목의 "활성화(Enable)"를 클릭한다.  6. 활성화(Enable) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 DynamoDB 테이블의 "PointInTimeRecoveryStatus" 항목을 "ENABLED"로 설정한다.  # aws dynamodb update-continuous-backups --table-name <table name> --point-in-time-recovery-specification PointInTimeRecoveryEnabled=true | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/dynamodb/ 를 통해 AWS DynamoDB에 접속한다.  2. 좌측 사이드 바에서 '테이블(Table)'을 클릭한다. 3. 각 테이블을 클릭한다.  4. 개요(Overview) 탭을 선택한다.  5. 테이블 세부 정보(Table details)에서 '특정 시점으로 복구(Point-in-time recovery)' 항목이 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 DynamoDB 테이블의 "PointInTimeRecoveryStatus" 항목이 "ENABLED"로 설정되어 있는지 확인한다.  # aws dynamodb describe-continuous-backups --table-name <table name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/PointInTimeRecovery.html https://docs.aws.amazon.com/config/latest/developerguide/dynamodb-pitr-enabled.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/PointInTimeRecovery.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-dynamodb-2 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | DynamoDB 리소스 태그 지정(4.3.4) | 위험도 | Low |
| 대분류 | Database | | |
| 중분류 | DynamoDB | | |
| 요약 | 각 DynamoDB 리소스에 태그를 지정하는 것을 권장한다. | | |
| 설명 | 태그는 AWS 리소스에 할당하는 키-값 쌍이다. 태그를 통해 리소스를 쉽게 관리 및 식별할 수 있으며, 작업을 허용하는 IAM 정책을 생성하여 액세스를 제어할 수 있다. 또한, AWS 청구서를 태그 별로 구분하여 분석할 수 있다.  ※ 태그가 지정된 테이블과 관련된 로컬 보조 인덱스(LSI) 및 글로벌 보조 인덱스(GSI)는 동일한 태그로 레이블이 자동 지정된다.  ※ 사용자 지정 태그 식별자는 대/소문자를 구분한다. | | |
| 진단 기준 | ▶ 양호 : 각 DynamoDB 리소스에 태그가 지정되어 있는 경우   ▶ 취약 : 각 DynamoDB 리소스에 태그가 지정되어 있지 않은 경우  ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/dynamodb/ 를 통해 AWS DynamoDB에 접속한다.  2. 좌측 사이드 바에서 '테이블(Table)'을 클릭한다. 3. 해당 테이블을 클릭한다.  4. 태그(Tags) 탭을 선택한다.  5. 태그 추가(Add Tags)에 키와 값을 입력한 후, 변경 사항 적용(Apply Changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 DynamoDB 리소스에 태그를 지정한다.  # aws dynamodb tag-resource --resource-arn <dynamodb arn> --tags Key=<key>,Value=<value> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/dynamodb/ 를 통해 AWS DynamoDB에 접속한다.  2. 좌측 사이드 바에서 '테이블(Table)'을 클릭한다. 3. 각 테이블을 클릭한다.  4. 태그(Tags) 탭을 선택한다.  5. 적용된 태그(Applied Tags)가 존재하는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 DynamoDB 리소스에 태그가 지정되어 있는지 확인한다.  # aws dynamodb list-tags-of-resource --resource-arn <dynamodb arn> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/best-practices-security-detective.html https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Tagging.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | DynamoDB 백업 계획 설정(4.3.5) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | DynamoDB | | |
| 요약 | AWS Backup 백업 계획을 통해 DynamoDB 테이블을 정기적으로 백업하고, 일정 기간 이상 보존하는 것을 권장한다. | | |
| 설명 | AWS Backup은 클라우드 및 온프레미스의 AWS 서비스 전반에서 데이터 백업을 쉽게 중앙집중화 및 자동화할 수 있는 서비스이며, 백업 관리를 간소화하고 비즈니스 및 규제 백업 규정 준수 요구사항을 충족할 수 있다. 백업 계획을 통해 백업 일정, 백업 보존 규칙 및 수명 주기 규칙을 비롯한 백업 요구 사항을 정의하여, 해당 계획에 따라 리소스를 자동으로 백업하고 유지할 수 있다. 기본 백업 기간은 오전 5시(UTC)에 시작하여 8시간 동안 지속된다. 매일(24시간마다) 백업하도록 설정하는 것을 권장한다.   백업 계획을 생성하는 방법은 세 가지가 있다.  - 템플릿으로 시작 : AWS Backup에서 제공하는 템플릿을 기반으로 백업 계획을 생성할 수 있다.   - Daily-35day-Retention (default) : 매일 백업 생성, 35일간 보존  - Daily-Monthly-1yr-Retention : 매일 및 매달 백업 생성, 1년간 보존  - Daily-Weekly-Monthly-5yr-Retention : 매일, 매주, 매달 백업 생성, 5년간 보존  - Daily-Weekly-Monthly-7yr-Retention : 매일, 매주, 매달 백업 생성, 7년간 보존 - 새 계획 수립 : 처음부터 새 백업 계획을 구성할 수 있다.  - JSON을 사용하여 계획 정의 : 기존 백업 계획의 JSON 표현식을 수정하거나 새 표현식을 생성할 수 있다. | | |
| 진단 기준 | ▶ 양호 : AWS Backup 백업 계획에 각 DynamoDB 테이블이 할당되어 정기적으로 백업이 수행되고 있는 경우   ▶ 취약 : AWS Backup 백업 계획에 각 DynamoDB 테이블이 할당되어 있지 않은 경우   ▶ Default Value : - ※ 리소스 자체 백업 기능을 통해 리소스를 백업하고 있는 경우, 해당 항목은 예외처리 할 수 있다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 백업 계획 생성(Create Backup plan) 버튼을 클릭한다.  4. 시작 옵션(Start options)에서 '템플릿으로 시작(Start with a template)'을 선택한다.  5. 템플릿 선택(Choose template)에서 템플릿 계획을 선택한다. 모든 템플릿에는 매일 백업하는 규칙(DailyBackups)이 적용되어 있다.  6. 백업 계획의 이름을 설정한 후, 계획 생성(Create plan) 버튼을 클릭한다.  ※ 시작 옵션에서 '새 계획 수립(Build a new plan)' 또는 'JSON을 사용하여 계획 정의(Define a plan using JSON)'를 선택하여 사용자가 직접 백업 계획을 구성할 수 있으며, 백업 빈도를 "매일(Daily)"로 설정하는 것을 권장한다. 7. 해당 백업 계획으로 이동한다.  8. 리소스 할당(Resource assignments)에서 "리소스 할당(Assign resources)" 버튼을 클릭한다.  9. 리소스 할당 이름 및 IAM 역할을 지정한 후, "다음을 기준으로 할당(Assign by)"에서 "리소스 ID(Resource ID)"를 선택한다.  10. "리소스 유형(Resource type)"에서 "DynamoDB"를 선택하고, "테이블 이름(Table name)"을 선택한다.  11. 리소스 할당(Assign resources) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 백업 계획을 생성한다.  # aws backup create-backup-plan --backup-plan <value>  예시는 다음과 같다.  # aws backup create-backup-plan --backup-plan "{\"BackupPlanName\":\"Example-Backup-Plan\",\"Rules\":[{\"RuleName\":\"DailyBackups\",\"ScheduleExpression\":\"cron(0 5 ? \* \* \*)\",\"StartWindowMinutes\":480,\"TargetBackupVaultName\":\"Default\",\"Lifecycle\":{\"DeleteAfterDays\":35}}]}"  2. 해당 백업 계획에 리소스를 할당한다.  # aws backup create-backup-selection --backup-plan-id <backup-plan-id> --backup-selection SelectionName="string",IamRoleArn="<iam role arn>",Resources="<resource arn>","<resource arn>",ListOfTags=[{ConditionType=string,ConditionKey=string,ConditionValue=string},{ConditionType=string,ConditionKey=string,ConditionValue=string}] | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/backup/ 를 통해 AWS Backup에 접속한다.  2. 좌측 사이드 바에서 '백업 계획(Backup plans)'을 클릭한다.  3. 각 백업 계획을 클릭한다.  4. 백업 규칙(Backup rules)이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다.  5. 리소스 할당(Resource assignments)에 DynamoDB 테이블이 지정되어 있는지 확인한다.  ※ 각 DynamoDB 테이블마다 설정된 백업 계획 및 백업 규칙은 다를 수 있다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 백업 계획의 백업 규칙이 사내 보안 규정에 맞도록 설정되어 있는지 확인한다. # aws backup get-backup-plan --backup-plan-id <backup-plan-id>  2. 백업 계획에 각 DynamoDB 테이블이 할당되어 있는지 확인한다.  # aws backup list-backup-selections --backup-plan-id <backup-plan-id> # aws backup get-backup-selection --backup-plan-id <backup-plan-id> --selection-id <selection id> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/config/latest/developerguide/dynamodb-in-backup-plan.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/cli/latest/reference/backup/create-backup-plan.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ElastiCache Redis 클러스터 자동 백업 설정(4.4.1) | 위험도 | Medium |
| 대분류 | Database | | |
| 중분류 | ElastiCache | | |
| 요약 | ElastiCache Redis 클러스터의 자동 백업을 설정하여 정기적으로 백업하는 것을 권장한다. | | |
| 설명 | Amazon ElastiCache Redis 클러스터에 자동 백업을 활성화할 경우, 클러스터 백업을 생성하여 장애가 발생했을 때 데이터를 복원할 수 있다. 백업 기간을 설정하여 사용자가 원하는 시간에 백업하도록 지정할 수 있으며, S3 버킷에 저장되는 백업 보존 일수를 지정할 수 있다. 최대 백업 보존 기간은 35일이며, 0으로 설정할 경우 클러스터의 자동 백업이 비활성화된다.  ※ Redis 클러스터의 경우, cache.t1.micro 노드 외의 모든 캐시 노드 유형에서 백업 및 복원을 지원한다. | | |
| 진단 기준 | ▶ 양호 : 각 ElastiCache Redis 클러스터에 자동 백업(Automatic backups)이 활성화되어 있는 경우   ▶ 취약 : 각 ElastiCache Redis 클러스터에 자동 백업이 비활성화되어 있는 경우   ▶ Default Value : 자동 백업 활성화, 백업 보존 기간 1일 | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/elasticache/ 를 통해 Amazon ElastiCache 대시보드에 접속한다.  2. 좌측 사이드 바에서 'Redis' 를 클릭한다.  3. 해당 Redis 클러스터를 체크한다.  4. 상단의 '작업(Actions)' 버튼을 클릭하여, '수정(Modify)'을 선택한다.  5. "자동 백업 활성화(Enable Automatic Backups)" 항목에 체크한 후, "백업 보존 기간(Backup Retention Period)"을 1일 이상으로 설정한다.  6. 수정(Modify) 버튼을 클릭한다.   [AWS CLI] ※ 자동 백업이 이미 활성화된 상태에서 백업 보존 기간을 수정하려는 경우  1. 다음 명령어를 실행하여 ElastiCache 클러스터의 백업 보존 기간을 수정한다.  # aws elasticache modify-replication-group --replication-group-id <replication group id> --snapshot-retention-limit <value>  ※ 자동 백업이 비활성화된 상태에서 자동 백업을 활성화하고 백업 보존 기간을 수정하려는 경우 1. 다음 명령어를 실행하여 ElastiCache 클러스터의 자동 백업을 활성화하고, 백업 보존 기간을 수정한다.  # aws elasticache modify-replication-group --replication-group-id <replication group id> --snapshot-retention-limit <value> --snapshotting-cluster-id <value>  ※ replication group id는 클러스터 이름을 의미한다.  ※ snapshot-retention-limit의 <value>에는 0~5 사이의 값을 넣을 수 있으며, 자동 백업을 활성화하기 위해서는 1~5 사이의 값을 입력해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/elasticache/ 를 통해 Amazon ElastiCache 대시보드에 접속한다.  2. 좌측 사이드 바에서 'Redis' 를 클릭한다.  3. 각 Redis 클러스터를 클릭한다.  4. 설명(Description) 탭을 선택한다.  5. 클러스터 세부 정보(Cluster Details) 에서 "백업 보존 기간(Backup Retention Period)" 항목이 1일 이상으로 설정되어 있는지 확인한다.  ※ 백업 보존 기간이 0으로 설정된 경우, 자동 백업이 비활성화되어 있음을 의미한다.  ※ 백업 보존 기간을 수정할 경우, 해당 클러스터 내 모든 노드에 적용된다.   [AWS CLI] 1. 다음 명령어를 실행하여 "SnapshotRetentionLimit" 항목이 1일 이상으로 설정되어 있는지 확인한다.  # aws elasticache describe-replication-groups --replication-group-id <replication group id>  ※ replication group id는 클러스터 이름을 의미한다.  ※ "SnapshotRetentionLimit" 항목이 0으로 설정된 경우, 자동 백업이 비활성화되어 있음을 의미한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/AmazonElastiCache/latest/red-ug/backups.html https://docs.aws.amazon.com/config/latest/developerguide/elasticache-redis-cluster-automatic-backup-check.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html | | |

**Logging**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudTrail 로그 파일 검증 활성화(5.1.2) | 위험도 | Medium |
| 대분류 | Logging | | |
| 중분류 | CloudTrail | | |
| 요약 | 로그 파일 검증을 사용할 경우 CloudTrail 로그에 대한 추가적인 무결성 검사가 가능하므로 해당 기능을 설정하는 것을 권장한다. | | |
| 설명 | CloudTrail 로그 파일 검증은 CloudTrail이 S3에 쓰는 각 로그의 해시를 포함한 전자 서명 다이제스트 파일을 생성한다. 다이제스트 파일을 통해 CloudTrail이 로그를 전달한 후 로그 파일이 변경, 삭제, 변경되지 않았는지 여부를 확인할 수 있다. CloudTrail 로그에 대한 추가적인 무결성 검사를 위해 로그 파일 검증을 사용하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : CloudTrail의 각 추적에 "로그 파일 검증(Log file validation)" 항목이 활성화되어 있는 경우   ▶ 취약 : CloudTrail의 각 추적에 "로그 파일 검증" 항목이 비활성화되어 있는 경우   ▶ Default Value : Enabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail 대시보드에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 해당 추적을 클릭한다. 4. 일반 세부 정보(General details) 우측의 '편집(Edit)'을 클릭한다. 5. 추가 설정(Additional settings) 항목 중 '로그 파일 검증(Log file validation)'의 "활성화됨(Enabled)"을 선택한 후, 변경 내용 저장(Save changes)을 클릭한다.  [AWS CLI] 1. 다음 명령어를 실행하여 로그의 유효성 검사를 활성화 한다. # aws cloudtrail update-trail --name <trail\_name> --enable-log-file-validation # aws cloudtrail validate-logs --trail-arn <trail\_arn> --start-time <start\_time> --end-time <end\_time> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 각 추적을 클릭한다. 4. '로그 파일 검증(Log file validation)'이 "활성화됨(Enabled)" 인지 확인한다.  [AWS CLI] 1. 각 추적마다 "LogFileValidationEnabled" 항목이 "true"로 설정되어 있는지 확인한다. # aws cloudtrail describe-trails | | |
| Reference | 3.2 Ensure CloudTrail log file validation is enabled (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-log-file-validation-enabling.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 정책 및 ACL의 퍼블릭 액세스 차단 설정(5.1.3) | 위험도 | High |
| 대분류 | Logging | | |
| 중분류 | CloudTrail | | |
| 요약 | CloudTrail이 기록하는 S3에 버킷 정책 또는 ACL을 적용하는 것을 권장한다. | | |
| 설명 | CloudTrail은 AWS 계정에서 이루어지는 모든 API 호출 이력을 기록하며, 해당 로그 파일은 S3 버킷에 저장된다. CloudTrail 로그에 대한 퍼블릭 액세스를 허용할 경우, 악의적인 사용자가 계정의 사용 또는 리소스 설정 값에 대한 취약점을 식별하는데 도움을 줄 수 있으므로 특정 사용자 외에 로그 파일에 접근하지 못하도록 버킷 정책과 ACL을 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : CloudTrail의 로그가 저장되는 S3 버킷 정책 내에 'Effect'가 "Allow"이고, 'Principal'이 "\*" or {"AWS" : "\*"}로 설정되어 있는 구문이 없는 경우   ▶ 취약 : CloudTrail의 로그가 저장되는 S3 버킷 정책 내에 'Effect'가 "Allow"이고, 'Principal'이 "\*" or {"AWS" : "\*"}로 설정되어 있는 구문이 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 추적의 'S3 버킷(S3 bucket)' 이름을 기록한다.  4. https://console.aws.amazon.com/s3/를 통해 Amazon S3에 접속한다. 5. 해당 이름의 S3 버킷을 클릭한다.  6. 권한(Permissions) 탭을 클릭한다.  7. ACL(액세스 제어 목록)(Access control list (ACL)) 우측의 '편집(Edit)' 버튼을 클릭한다.  8. "모든 사용자(Everyone)" 또는 "인증된 사용자(Authenticated users)"에게 부여된 권한을 모두 해제한다.  9. 변경 사항 저장(Save changes) 버튼을 클릭한다.  10. 버킷 정책(Bucket policy) 우측의 '편집(Edit)' 버튼을 클릭한다.  11. 'Effect' 가 "Allow" 이고, 'Principal' 이 "\*" or {"AWS" : "\*"} 로 설정되어 있는 구문을 삭제한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 추적의 'S3 버킷(S3 bucket)' 이름을 기록한다.  4. https://console.aws.amazon.com/s3/를 통해 Amazon S3에 접속한다. 5. 해당 이름의 S3 버킷을 클릭한다.  6. 권한(Permissions) 탭을 클릭한다.  7. ACL(액세스 제어 목록)(Access control list (ACL))에서 '피부여자(Grantee)'가 "모든 사용자(Everyone)" 또는 "인증된 사용자(Authenticated users)"로 설정되어 있는지 확인한다.  8. 버킷 정책(Bucket policy) 우측의 '편집(Edit)' 버튼을 클릭한다.  9. 'Effect' 가 "Allow" 이고, 'Principal' 이 "\*" or {"AWS" : "\*"} 로 설정되어 있는 구문이 있는지 확인한다.   [AWS CLI] 1. 추적 로그 버킷 이름을 확인한다. # aws cloudtrail describe-trails --query 'trailList[\*].S3BucketName'   2. AllUsers principal이 <bucket>에 특권이 할당되어 있지 않은지 확인한다. # aws s3api get-bucket-acl --bucket <s3\_bucket\_for\_cloudtrail> --query 'Grants[?Grantee.URI== `https://acs.amazonaws.com/groups/global/AllUsers` ]'   3. AuthenticatedUsers principal이 <bucket>에 특권이 할당되어 있지 않은지 확인한다.  # aws s3api get-bucket-acl --bucket <s3\_bucket\_for\_cloudtrail> --query 'Grants[?Grantee.URI== `https://acs.amazonaws.com/groups/global/Authenticated Users` ]'   4. S3 버킷 정책을 확인한다. # aws s3api get-bucket-policy --bucket <s3\_bucket\_for\_cloudtrail>   5. 구문에 'Effect' 가 "Allow" 이고, 'Principal' 이 "\*" 또는 {"AWS" : "\*"}로 설정되어 있는지 확인한다. ※ Principal이 "\*" 또는 {"AWS" : "\*"} 로 설정되어 있을 경우, 모든 사용자가 접근 가능하다는 것을 의미한다. | | |
| Reference | 3.3 Ensure the S3 bucket used to store CloudTrail logs is not publicly accessible (Automated) | | |
| Site | https://docs.aws.amazon.com/IAM/latest/UserGuide/reference\_policies\_elements\_principal.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudTrail 추적과 Cloudwatch Logs 통합 설정(5.1.4) | 위험도 | Medium |
| 대분류 | Logging | | |
| 중분류 | CloudTrail | | |
| 요약 | 계정의 모든 리전에서 추적을 활성화하여 CloutTrail 로그를 CloudWatch Logs로 보내는 것을 권장한다. | | |
| 설명 | CloudTrail은 로그 파일 저장 및 전송에 S3를 사용하기 때문에 로그 파일을 장기간 저장할 수 있다. 장기 분석을 위해 지정된 S3 버킷 내에서 CloudTrail 로그 파일을 캡쳐하는 것 외에도 CloudWatch Logs로 로그를 전송하여 실시간 분석을 수행할 수 있다. CloudTrail 로그를 CloudWatch Logs로 전송할 경우, 사용자, API, 리소스 및 IP 주소를 기반으로 실시간 및 활동 기록 로깅이 가능하며, 비정상적이거나 민감한 계정 활동에 대한 경보 및 알림을 설정할 수 있다. AWS 계정 활동을 모니터링하고 경고 알람을 위해 CloudTrail 로그를 CloudWatch Logs로 보내는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : CloudTrail의 각 추적에서 'CloudWatch Logs' 항목의 로그 그룹과 IAM 역할이 설정되어 있는 경우   ▶ 취약 : CloudTrail의 각 추적에서 'CloudWatch Logs' 항목의 로그 그룹과 IAM 역할이 설정되어 있지 않은 경우  ▶ Default Value : CloudWatch Logs가 설정되어 있지 않음 | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail 대시보드에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 해당 추적을 클릭한다. 4. 'CloudWatch Logs' 우측의 '편집(Edit)'을 클릭한다. 5. 'CloudWatch Logs' 항목에서 "활성화됨(Enabled)"을 선택한 후, 로그 그룹(Log group)과 IAM 역할(IAM Role)을 설정한다. 6. 변경 내용 저장(Save changes)을 클릭한다.  [AWS CLI] 1. 다음 명령어를 실행하여 추적을 업데이트한다.  # aws cloudtrail update-trail --name <trail\_name> --cloudwatch-logs-log-group-arn <cloudtrail\_log\_group\_arn> --cloudwatch-logs-role-arn <cloudtrail\_cloudwatchLogs\_role\_arn> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 각 추적을 클릭한다. 4. 'CloudWatch Logs' 항목의 "로그 그룹(Log group)"과 "IAM 역할(IAM Role)"이 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 추적이 존재하는지 확인한다. "CloudWatchLogsLogGroupArn" 값이 비어있지 않아야 한다.  # aws cloudtrail describe-trails  2. 각 추적의 "LatestCloudWatchLogsDeliveryTime" 속성 값이 현재 타임스탬프로 설정되어 있는지 확인한다. # aws cloudtrail get-trail-status --name <trail\_name> | | |
| Reference | 3.4 Ensure CloudTrail trails are integrated with CloudWatch Logs (Automated) | | |
| Site | https://aws.amazon.com/ko/cloudtrail/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudTrail S3 버킷의 서버 액세스 로깅 활성화(5.1.6) | 위험도 | Medium |
| 대분류 | Logging | | |
| 중분류 | CloudTrail | | |
| 요약 | CloudTrail S3 버킷에서 서버 액세스 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | S3 버킷의 서버 액세스 로깅은 버킷에 대한 액세스 요청을 기록한다. 액세스 로그에는 요청 유형, 작동된 요청에 지정된 리소스, 요청이 처리된 시간 및 날짜와 같은 세부 정보가 포함된다. 서버 액세스 로깅을 활성화할 경우, 버킷 내의 객체에 영향을 줄 수 있는 모든 이벤트를 캡처할 수 있다. 로그를 별도의 버킷에 배치하도록 구성할 경우, 보안 및 문제 대응 프로세스에 유용한 로그 정보를 얻을 수 있다. | | |
| 진단 기준 | ▶ 양호 : 추적 버킷에 "서버 액세스 로깅(Server access logging)"이 활성화되어 있는 경우   ▶ 취약 : 추적 버킷에 "서버 액세스 로깅"이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://s3.console.aws.amazon.com/s3/home 를 통해 Amazon S3에 접속한다. 2. 해당 버킷을 클릭한다. 3. 속성(Properties) 탭을 클릭한다.  4. 서버 액세스 로깅(Server access logging) 항목의 '편집(Edit)' 버튼을 클릭한다.  5. 서버 액세스 로깅 항목에서 "활성화(Enable)"를 클릭하고, 대상 버킷을 선택한다.  6. 변경 사항 저장(Save changes) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 각 추적을 클릭한다.  4. 추적의 로그가 저장되는 버킷으로 이동한다.  6. 속성(Properties) 탭을 클릭한다.  7. '서버 액세스 로깅(Server access logging)' 항목이 "활성화됨(Enabled)" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 해당 추적의 추적 로그 위치("S3BucketName")를 확인한다.  # aws cloudtrail get-trail --name <trail name>  2. 서버 액세스 로깅이 활성화되어 있는지 확인한다. 결과 값이 비어있는 경우 비활성화를 의미한다. # aws s3api get-bucket-logging --bucket <s3 bucket name>  예시는 다음과 같다.  {  "LoggingEnabled": {  "TargetBucket": "<s3 bucket name>",  "TargetPrefix": ""  } } | | |
| Reference | 3.6 Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket (Automated) | | |
| Site | https://docs.aws.amazon.com/AmazonS3/latest/dev/ServerLogs.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-logging-enabled.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudTrail 로그 보관 시 고객 마스터 키로 암호화(5.1.7) | 위험도 | Medium |
| 대분류 | Logging | | |
| 중분류 | CloudTrail | | |
| 요약 | KMS 고객 마스터 키를 사용하여 CloudTrail 로그 파일을 암호화하는 것을 권장한다. | | |
| 설명 | AWS CloudTrail은 계정에 대한 AWS API 호출을 기록하고 IAM 정책에 따라 사용자와 리소스가 해당 로그를 생성할 수 있도록 하는 서비스이다. 기록 정보에는 API를 호출한 사용자의 ID, API 호출 시간, API 호출한 사용자의 IP 주소, 요청 매개 변수 및 AWS 서비스에서 반환된 응답 요소 등이 포함된다. AWS KMS(Key Management Service)는 계정 데이터 암호화에 사용되는 암호화 키를 생성 및 제어하는 관리형 서비스로 암호화 키의 보안을 위해 HSM(Hardware Security Module)을 사용한다. CloudTrail 로그는 SSE(Server Side Encryption) 및 KMS 고객 마스터 키(CMK, Customer master key)를 활용하여 보다 높은 보안을 제공한다. SSE-KMS를 사용하여 로그를 암호화할 경우, 해당 로그 버킷에 대한 S3 읽기 권한이 있어야 하며 CMK 정책에 의해 복호화 권한이 부여되어 있어야 한다. 이러한 권한이 필요하므로 설정 시 기밀성을 높일 수 있다. | | |
| 진단 기준 | ▶ 양호 : CloudTrail의 각 추적에 "로그 파일 SSE-KMS 암호화(Log file SSE-KMS encryption)" 항목이 활성화되어 있는 경우   ▶ 취약 : CloudTrail의 각 추적에 "로그 파일 SSE-KMS 암호화" 항목이 비활성화되어 있는 경우   ▶ Default Value : Enabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 해당 추적을 클릭한다. 4. 일반 세부 정보(General details)의 '편집(Edit)'을 클릭한다. 5. '로그 파일 SSE-KMS 암호화(Log file SSE-KMS encryption)' 항목에서 "활성화됨(Enabled)"을 선택한다. 6. 'AWS KMS 고객 관리형 CMK(AWS KMS customer managed CMK)'를 선택한 후, AWS KMS 별칭(AWS KMS alias)을 입력한다. 7. 변경 내용 저장(Save changes)을 클릭한다.  [AWS CLI] 1. 다음 명령어를 실행하여 각 추적의 SSE-KMS를 활성화한다. # aws cloudtrail update-trail --name <trail\_name> --kms-id <cloudtrail\_kms\_key>  # aws kms put-key-policy --key-id <cloudtrail\_kms\_key> --policy <cloudtrail\_kms\_key\_policy> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudtrail/ 를 통해 AWS CloudTrail에 접속한다. 2. 좌측 사이드 바에서 '추적(Trails)'을 클릭한다.  3. 각 추적을 클릭한다. 4. 일반 세부 정보(General details)에서 '로그 파일 SSE-KMS 암호화(Log file SSE-KMS encryption)' 항목이 "활성화됨(Enabled)" 인지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 각 추적의 "KmsKeyId" 속성값이 정의되어 있는지 확인한다.  # aws cloudtrail describe-trails  ※ 정의된 경우 SSE-KMS가 활성화됨을 의미한다. | | |
| Reference | 3.7 Ensure CloudTrail logs are encrypted at rest using KMS CMKs (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/encrypting-cloudtrail-log-files-with-aws-kms.html https://docs.aws.amazon.com/kms/latest/developerguide/create-keys.html | | |

**Monitoring**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 인증되지 않은 API 호출에 대한 로그 지표 및 경보 설정(6.1.1) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 인증되지 않은 API 호출에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 인증되지 않은 API 호출에 대해 모니터링할 경우, 응용 프로그램 오류를 확인할 수 있으며 악의적인 활동을 탐지하는데 걸리는 시간을 줄일 수 있다. 과도한 수의 경고가 생성되는 경우 조직은 제한된 IAM 사용자 권한에 읽기 접근 권한을 추가하여 경고를 줄이는 방법을 고려할 수 있다. 경우에 따라 사용자가 실제 시스템의 일부 영역을 볼 수도 있으므로 할당된 추가 액세스 권한이 기존 제한된 IAM 사용자의 의도와 일치하는지 검토해야 한다. | | |
| 진단 기준 | ▶ 양호 : 인증되지 않은 API 호출에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 인증되지 않은 API 호출에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.errorCode = "\*UnauthorizedOperation") || ($.errorCode = "AccessDenied\*") || ($.sourceIPAddress!="delivery.logs.amazonaws.com") || ($.eventName!="HeadBucket")}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail 내 모든 추적을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 들 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나 이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. “filterPattern” = “{(($.errorCode=“UnauthorizedOperation”) || ($.errorCode=“AccessDenied”)) || (($.sourceIPAddress!=“delivery.logs.amazonaws.com”) || ($.eventName!=“HeadBucket”))}”  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.1 Ensure a log metric filter and alarm exist for unauthorized API calls (Automated) | | |
| Site | https://aws.amazon.com/ko/sns/?whats-new-cards.sort-by=item.additionalFields.postDateTime&whats-new-cards.sort-order=desc https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/CreateMetricFilterProcedure.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | MFA 없이 관리 콘솔 로그인에 대한 로그 지표 및 경보 설정(6.1.2) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | MFA 없이 관리 콘솔 로그인에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 단일 요인 콘솔 로그인(MFA를 사용하지 않는 경우)을 모니터링할 경우 MFA에 의해 보호되지 않는 계정에 대한 가시성을 향상시킬 수 있다. | | |
| 진단 기준 | ▶ 양호 : MFA 없이 관리 콘솔 로그인에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : MFA 없이 관리 콘솔 로그인에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)' 을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = "ConsoleLogin") && ($.additionalEventData.MFAUsed != "Yes")}  ※ 조직에서 SSO를 사용할 경우 패턴 필터링을 {($.eventName = "ConsoleLogin") && ($.additionalEventData.MFAUsed != "Yes") && ($.userIdentity.type = "IAMUser") && ($.responseElements.ConsoleLogin = "Success")}로 입력하는 것을 권장한다.  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail 내 모든 추적을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 들 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나 이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern" = "{($.eventName = "ConsoleLogin") && ($.additionalEventData.MFAUsed != "Yes")}”  ※ 조직에서 SSO를 사용할 경우 필터 값 중 다음 값이 있는지 확인한다.  "filterPattern" = "{($.eventName = "ConsoleLogin") && ($.additionalEventData.MFAUsed != "Yes") && ($.userIdentity.type = "IAMUser") && ($.responseElements.ConsoleLogin = "Success")}“  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.2 Ensure a log metric filter and alarm exist for Management Console sign-in without MFA (Automated) | | |
| Site | https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/viewing\_metrics\_with\_cloudwatch.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 루트 계정 사용에 대한 로그 지표 및 경보 설정(6.1.3) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 루트 로그인 시도에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 루트 계정 로그인을 모니터링할 경우, 중요 계정 사용에 대한 가시성을 향상시킬 수 있다. | | |
| 진단 기준 | ▶ 양호 : 루트 로그인 시도에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 루트 로그인 시도에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {$.userIdentity.type = "Root" && $.userIdentity.invokedBy NOT EXISTS && $.eventType != "AwsServiceEvent"}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 들 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern” = "{$.userIdentity.type = "Root" && $.userIdentity.invokedBy NOT EXISTS && $.eventType != "AwsServiceEvent"}”  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.3 Ensure a log metric filter and alarm exist for usage of 'root' account (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | IAM 정책 변경사항에 대한 로그 지표 및 경보 설정(6.1.4) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | IAM 정책 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. IAM 정책에 대한 변경사항을 모니터링할 경우, 인증 및 권한 제어의 기존 설정 값이 유지되도록 하는 데 도움이 된다. | | |
| 진단 기준 | ▶ 양호 : IAM 정책 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : IAM 정책 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)' 을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName=DeleteGroupPolicy) || ($.eventName=DeleteRolePolicy) || ($.eventName=DeleteUserPolicy) || ($.eventName=PutGroupPolicy) || ($.eventName=PutRolePolicy) || ($.eventName=PutUserPolicy) || ($.eventName=CreatePolicy) || ($.eventName=DeletePolicy) || ($.eventName=CreatePolicyVersion) || ($.eventName=DeletePolicyVersion) || ($.eventName=AttachRolePolicy) || ($.eventName=DetachRolePolicy) || ($.eventName=AttachUserPolicy) || ($.eventName=DetachUserPolicy) || ($.eventName=AttachGroupPolicy) || ($.eventName=DetachGroupPolicy)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail" 가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 들 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName=DeleteGroupPolicy)||($.eventName=DeleteRolePolicy)||($.eventName=DeleteUserPolicy)||($.eventName=PutGroupPolicy)||($.eventName=PutRolePolicy)||($.eventName=PutUserPolicy)||($.eventName=CreatePolicy)||($.eventName=DeletePolicy)||($.eventName=CreatePolicyVersion)||($.eventName=DeletePolicyVersion)||($.eventName=AttachRolePolicy)||($.eventName=DetachRolePolicy)||($.eventName=AttachUserPolicy)||($.eventName=DetachUserPolicy)||($.eventName=AttachGroupPolicy)||($.eventName=DetachGroupPolicy)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.4 Ensure a log metric filter and alarm exist for IAM policy changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudTrail 설정 변경사항에 대한 로그 지표 및 경보 설정(6.1.5) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | CloudTrail 설정 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. CloudTrail 설정 변경사항을 모니터링할 경우, AWS 계정에서 수행된 활동을 지속적으로 확인할 수 있다. | | |
| 진단 기준 | ▶ 양호 : CloudTrail 설정 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : CloudTrail 설정 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)' 을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)' 을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = CreateTrail) || ($.eventName = UpdateTrail) || ($.eventName = DeleteTrail) || ($.eventName = StartLogging) || ($.eventName = StopLogging)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail> 은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = CreateTrail) || ($.eventName = UpdateTrail) || ($.eventName = DeleteTrail) || ($.eventName = StartLogging) || ($.eventName = StopLogging)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.5 Ensure a log metric filter and alarm exist for CloudTrail configuration changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | AWS 관리 콘솔 인증 실패에 대한 로그 지표 및 경보 설정(6.1.6) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | AWS 관리 콘솔 인증 실패에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 관리 콘솔 로그인 실패를 모니터링할 경우, 무차별 대입 공격을 탐지하는 데 걸리는 시간을 줄일 수 있다. 또한 다른 이벤트 상관 관계에서 사용할 수 있는 출발지 IP와 같은 식별자를 제공한다. | | |
| 진단 기준 | ▶ 양호 : AWS 관리 콘솔 인증 실패에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : AWS 관리 콘솔 인증 실패에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = ConsoleLogin) && ($.errorMessage = "Failed authentication")}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail> 은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails> 은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = ConsoleLogin) && ($.errorMessage = "Failed authentication")}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.6 Ensure a log metric filter and alarm exist for AWS Management Console authentication failures (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 사용자가 생성한 CMK 비활성화 또는 삭제 예약에 대한 로그 지표 및 경보 설정(6.1.7) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 사용자가 생성한 CMK 비활성화 또는 삭제 예약에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 비활성화되거나 삭제된 키로 암호화된 데이터는 더 이상 접근할 수 없다. 해당 키에 대한 경보를 통해 비활성화되거나 삭제된 키로 암호화하지 않도록 경보를 설정하고 사전에 방지해야 한다. | | |
| 진단 기준 | ▶ 양호 : 사용자가 생성한 CMK 비활성화 또는 삭제 예약에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 사용자가 생성한 CMK 비활성화 또는 삭제 예약에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventSource = kms.amazonaws.com) && (($.eventName=DisableKey)||($.eventName=ScheduleKeyDeletion))}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail> 은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails> 은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventSource = kms.amazonaws.com) && (($.eventName=DisableKey)||($.eventName=ScheduleKeyDeletion))}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.7 Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer created CMKs (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | S3 버킷 정책 변경사항에 대한 로그 지표 및 경보 설정(6.1.8) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | S3 버킷 정책 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. S3 버킷 정책 변경사항을 모니터링할 경우, 민감정보가 담긴 S3 버킷에 허용 정책을 검색 및 수정하는 데 걸리는 시간을 줄일 수 있다. | | |
| 진단 기준 | ▶ 양호 : S3 버킷 정책 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : S3 버킷 정책 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventSource = s3.amazonaws.com) && (($.eventName = PutBucketAcl) || ($.eventName = PutBucketPolicy) || ($.eventName = PutBucketCors) || ($.eventName = PutBucketLifecycle) || ($.eventName = PutBucketReplication) || ($.eventName = DeleteBucketPolicy) || ($.eventName = DeleteBucketCors) || ($.eventName = DeleteBucketLifecycle) || ($.eventName = DeleteBucketReplication))}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventSource = s3.amazonaws.com) && (($.eventName = PutBucketAcl) || ($.eventName = PutBucketPolicy) || ($.eventName = PutBucketCors) || ($.eventName = PutBucketLifecycle) || ($.eventName = PutBucketReplication) || ($.eventName = DeleteBucketPolicy) || ($.eventName = DeleteBucketCors) || ($.eventName = DeleteBucketLifecycle) || ($.eventName = DeleteBucketReplication))}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.8 Ensure a log metric filter and alarm exist for S3 bucket policy changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | AWS Config 설정 변경사항에 대한 로그 지표 및 경보 설정(6.1.9) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | AWS Config 설정 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. AWS Config 설정 변경사항을 모니터링할 경우, AWS 계정 내의 설정 변경 값을 지속적으로 확인할 수 있다. | | |
| 진단 기준 | ▶ 양호 : AWS Config 설정 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : AWS Config 설정 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventSource = config.amazonaws.com) && (($.eventName=StopConfigurationRecorder)||($.eventName=DeleteDeliveryChannel) ||($.eventName=PutDeliveryChannel)||($.eventName=PutConfigurationRecorder))}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중 인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventSource = config.amazonaws.com) && (($.eventName=StopConfigurationRecorder)||($.eventName=DeleteDeliveryChannel) ||($.eventName=PutDeliveryChannel)||($.eventName=PutConfigurationRecorder))}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.9 Ensure a log metric filter and alarm exist for AWS Config configuration changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 보안 그룹 변경사항에 대한 로그 지표 및 경보 설정(6.1.10) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 보안 그룹 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 보안 그룹 변경사항을 모니터링할 경우, 리소스 및 서비스가 의도치 않게 노출되는 것을 방지할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 보안 그룹 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 보안 그룹 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)' 을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = AuthorizeSecurityGroupIngress) || ($.eventName = AuthorizeSecurityGroupEgress) || ($.eventName = RevokeSecurityGroupIngress) || ($.eventName = RevokeSecurityGroupEgress) || ($.eventName = CreateSecurityGroup) || ($.eventName = DeleteSecurityGroup)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = AuthorizeSecurityGroupIngress) || ($.eventName = AuthorizeSecurityGroupEgress) || ($.eventName = RevokeSecurityGroupIngress) || ($.eventName = RevokeSecurityGroupEgress) || ($.eventName = CreateSecurityGroup) || ($.eventName = DeleteSecurityGroup)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.10 Ensure a log metric filter and alarm exist for security group changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 네트워크 ACL 변경사항에 대한 로그 지표 및 경보 설정(6.1.11) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 네트워크 ACL 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 네트워크 ACL 변경사항을 모니터링할 경우, 리소스 및 서비스가 의도치 않게 노출되는 것을 방지할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 네트워크 ACL 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 네트워크 ACL 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = CreateNetworkAcl) || ($.eventName = CreateNetworkAclEntry) || ($.eventName = DeleteNetworkAcl) || ($.eventName = DeleteNetworkAclEntry) || ($.eventName = ReplaceNetworkAclEntry) || ($.eventName = ReplaceNetworkAclAssociation)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중 인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> \* <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 들 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = CreateNetworkAcl) || ($.eventName = CreateNetworkAclEntry) || ($.eventName = DeleteNetworkAcl) || ($.eventName = DeleteNetworkAclEntry) || ($.eventName = ReplaceNetworkAclEntry) || ($.eventName = ReplaceNetworkAclAssociation)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.11 Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL) (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 네트워크 게이트웨이 변경사항에 대한 로그 지표 및 경보 설정(6.1.12) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 네트워크 게이트웨이 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 트래픽을 VPC 외부의 목적지로 송수신하기 위해 네트워크 게이트웨이가 필요하다. 네트워크 게이트웨이 변경사항을 모니터링할 경우, 모든 수신/송신 트래픽이 제어된 경로를 통해 VPC 경계를 통과하는지 확인할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 네트워크 게이트웨이 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 네트워크 게이트웨이 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = CreateCustomerGateway) || ($.eventName = DeleteCustomerGateway) || ($.eventName = AttachInternetGateway) || ($.eventName = CreateInternetGateway) || ($.eventName = DeleteInternetGateway) || ($.eventName = DetachInternetGateway)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)"을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = CreateCustomerGateway) || ($.eventName = DeleteCustomerGateway) || ($.eventName = AttachInternetGateway) || ($.eventName = CreateInternetGateway) || ($.eventName = DeleteInternetGateway) || ($.eventName = DetachInternetGateway)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.12 Ensure a log metric filter and alarm exist for changes to network gateways (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 라우팅 테이블 변경사항에 대한 로그 지표 및 경보 설정(6.1.13) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 라우팅 테이블 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 라우팅 테이블은 서브넷과 네트워크 게이트웨이로 네트워크 트래픽을 라우팅하는데 사용된다. 라우팅 테이블 변경사항을 모니터링할 경우, 모든 VPC 트래픽이 예상 경로를 통해 이동하는지 확인할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 라우팅 테이블 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : 라우팅 테이블 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = CreateRoute) || ($.eventName = CreateRouteTable) || ($.eventName = ReplaceRoute) || ($.eventName = ReplaceRouteTableAssociation) || ($.eventName = DeleteRouteTable) || ($.eventName = DeleteRoute) || ($.eventName = DisassociateRouteTable)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)" 을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2.해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = CreateRoute) || ($.eventName = CreateRouteTable) || ($.eventName = ReplaceRoute) || ($.eventName = ReplaceRouteTableAssociation) || ($.eventName = DeleteRouteTable) || ($.eventName = DeleteRoute) || ($.eventName = DisassociateRouteTable)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.13 Ensure a log metric filter and alarm exist for route table changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | VPC 변경사항에 대한 로그 지표 및 경보 설정(6.1.14) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | VPC 변경사항에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail 로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. 한 계정 내에 2개 이상의 VPC가 존재할 수 있으며 네트워크 트래픽이 VPC 간에 라우팅되도록 피어링 연결을 생성할 수도 있다. VPC 변경사항을 모니터링할 경우, 모든 VPC 트래픽 흐름이 영향을 받지 않도록하는데 도움을 줄 수 있다. | | |
| 진단 기준 | ▶ 양호 : VPC 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : VPC 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)' 을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventName = CreateVpc) || ($.eventName = DeleteVpc) || ($.eventName = ModifyVpcAttribute) || ($.eventName = AcceptVpcPeeringConnection) || ($.eventName = CreateVpcPeeringConnection) || ($.eventName = DeleteVpcPeeringConnection) || ($.eventName = RejectVpcPeeringConnection) || ($.eventName = AttachClassicLinkVpc) || ($.eventName = DetachClassicLinkVpc) || ($.eventName = DisableVpcClassicLink) || ($.eventName = EnableVpcClassicLink)}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)" 을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. ※ <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> ※ <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> ※ <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> ※ <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventName = CreateVpc) || ($.eventName = DeleteVpc) || ($.eventName = ModifyVpcAttribute) || ($.eventName = AcceptVpcPeeringConnection) || ($.eventName = CreateVpcPeeringConnection) || ($.eventName = DeleteVpcPeeringConnection) || ($.eventName = RejectVpcPeeringConnection) || ($.eventName = AttachClassicLinkVpc) || ($.eventName = DetachClassicLinkVpc) || ($.eventName = DisableVpcClassicLink) || ($.eventName = EnableVpcClassicLink)}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. ※ AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  ※ SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.14 Ensure a log metric filter and alarm exist for VPC changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | AWS Organizations 변경사항에 대한 로그 지표 및 경보 설정(6.1.15) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | CloudWatch | | |
| 요약 | 마스터 AWS 계정에서 변경한 AWS Organizations에 대한 메트릭 필터 및 경보를 설정하는 것을 권장한다. | | |
| 설명 | API 호출에 대한 실시간 모니터링 설정은 CloudTrail로그를 CloudWatch Logs로 전송 및 해당 메트릭 필터와 경보를 설정하면 된다. AWS Organizations 변경사항을 모니터링할 경우, 무단 액세스 또는 기타 보안 침해로 이어질 수 있는 악의적인 수정을 방지할 수 있다. 또한, AWS Organizations 내 발생한 의도하지 않은 변경사항을 조사하여 해당 변경사항을 되돌릴 수 있다. | | |
| 진단 기준 | ▶ 양호 : AWS Organizations 변경사항에 대한 로그 지표 및 경보가 설정되어 있는 경우   ▶ 취약 : AWS Organizations 변경사항에 대한 로그 지표 및 경보가 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] ※ 지표 필터 생성 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 각 로그 그룹을 클릭한다.  ※ 로그 그룹이 존재하지 않는 경우, 우측 상단의 '로그 그룹 생성(Create log group)'을 클릭하여 새 로그 그룹을 생성한다. 보존 설정 기간(Retention setting)은 6개월 이내로 설정하는 것을 권장한다.  4. 우측 상단의 '작업' 버튼을 클릭하여, '지표 필터 생성(Create metric filter)'을 선택한다.  5. 패턴 필터링(Filter pattern) 항목에 아래 구문을 입력한 후, Next 버튼을 클릭한다.  {($.eventSource = organizations.amazonaws.com) && (($.eventName = "AcceptHandshake") || ($.eventName = "AttachPolicy") || ($.eventName = "CreateAccount") || ($.eventName = "CreateOrganizationalUnit") || ($.eventName = "CreatePolicy") || ($.eventName = "DeclineHandshake") || ($.eventName = "DeleteOrganization") || ($.eventName = "DeleteOrganizationalUnit") || ($.eventName = "DeletePolicy") || ($.eventName = "DetachPolicy") || ($.eventName = "DisablePolicyType") || ($.eventName = "EnablePolicyType") || ($.eventName = "InviteAccountToOrganization") || ($.eventName = "LeaveOrganization") || ($.eventName = "MoveAccount") || ($.eventName = "RemoveAccountFromOrganization") || ($.eventName = "UpdatePolicy") || ($.eventName = "UpdateOrganizationalUnit"))}  6. 필터 이름, 지표 네임스페이스, 지표 이름, 지표 값(1 이상)을 입력한 후, Next 버튼을 클릭한다.  7. 지표 필터 생성(Create metric filter) 버튼을 클릭한다.   ※ CloudTrail과 CloudWatch 연동 1. https://console.aws.amazon.com/cloudtrail 를 통해 AWS CloudTrail에 접속한다. 2. '추적 생성(Create trail)' 버튼을 클릭한다.  3. 추적 이름, 스토리지 위치, 암호화 여부 등을 설정한다.  4. CloudWatch Logs 항목은 "활성화됨(Enabled)"을 선택한다. 로그 그룹(Log group)은 "기존(Existing)" 을 클릭한 후, 지표 필터를 생성한 로그 그룹 이름을 입력한다.  5. IAM 역할을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  6. 이벤트 유형을 선택한 후, 하단의 다음(Next) 버튼을 클릭한다.  7. 추적 생성(Create trail) 버튼을 클릭한다.   ※ 알람 설정 1. https://console.aws.amazon.com/cloudwatch/ 를 통해 AWS CloudWatch에 접속한다. 2. 좌측 사이드 바에서 로그의 '로그 그룹(Log groups)'을 클릭한다.  3. 지표 필터를 생성한 로그 그룹을 클릭한다.  4. 지표 필터(Metric filters) 탭을 클릭한다.  5. 생성한 지표 필터를 체크한 후, '경보 생성(Create alarm)' 버튼을 클릭한다.  6. 지표 이름, 통계, 기간을 설정한다.  7. 임계값 유형(Threshold type)을 "정적(Static)", 경보 조건(alarm condition)은 "0 보다 큼(Greater)" 또는 "1 보다 크거나 같음(Greater/equal)"으로 설정한 후, 하단의 다음(Next) 버튼을 클릭한다.  8. 경보 상태 트리거(Alarm state trigger)에서 "경보 상태(In alarm)"로 설정한다. SNS 주제를 선택한 후, 다음(Next) 버튼을 클릭한다.  9. 경보 이름을 입력한 후, 다음(Next) 버튼을 클릭한다.  10. 경보 생성(Create alarm) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS CLI] 지정된 메트릭 필터 및 경보가 구성된 다중 리전 CloudTrail이 하나 이상 있는지 확인한다. 1. 활성화되어 있는 다중 리전 CloudTrail에 설정된 로그 그룹 이름을 확인한다. 1-1. CloudTrail을 나열한다. # aws cloudtrail describe-trails  1-2. 나열된 추적(Trails) 중 “IsMultiRegionTrail” 항목이 true 로 설정된 추적이 있는지 확인한다. 1-3. 해당 추적에서 “CloudWatchLogsLogGroupArn”의 <cloudtrail\_log\_group\_name>을 확인한다. \* <cloudtrail\_log\_group\_name>의 위치는 다음과 같다. "CloudWatchLogsLogGroupArn" : “arn:aws:logs:<region>:<aws\_account\_number>:log-group:<cloudtrail\_log\_group\_name>:\*“,  1-4. 다중 리전 CloudTrail이 로깅 중인지 확인한다. "IsLogging" 항목이 true 로 설정되어 있는지 확인한다.  # aws cloudtrail get-trail-status --name <Name of a Multi-region CloudTrail> \* <Name of a Multi-region CloudTrail>은 1-1번의 명령어를 통해 확인한 "IsMultiRegionTrail"가 true로 설정되어 있는 추적의 “Name”을 의미한다.  1-5. 다음 명령어를 통해 <Name of a Multi-region CloudTrail> 중에서 "ReadWriteType" 항목이 All 로 설정되어 있고, "IncludeManagementEvents" 항목이 true 로 설정되어 있는 추적이 하나이상 있는지 확인한다. # aws cloudtrail get-event-selectors --trail-name <trailname shown in describe-trails> \* <trailname shown in describe-trails>은 <Name of a Multi-region CloudTrail> 추적 중 하나를 의미한다.  2. 해당 로그 그룹에 지표 필터(metric filter)가 설정되어 있는지 확인한다. # aws logs describe-metric-filters --log-group-name <cloudtrail\_log\_group\_name> \* <cloudtrail\_log\_group\_name> 값은 1-3번에서 확인한 값이다.  3. 지표 필터 중 다음 값이 있는지 확인한다. "filterPattern": "{($.eventSource = organizations.amazonaws.com) && (($.eventName = "AcceptHandshake") || ($.eventName = "AttachPolicy") || ($.eventName = "CreateAccount") || ($.eventName = "CreateOrganizationalUnit") || ($.eventName = "CreatePolicy") || ($.eventName = "DeclineHandshake") || ($.eventName = "DeleteOrganization") || ($.eventName = "DeleteOrganizationalUnit") || ($.eventName = "DeletePolicy") || ($.eventName = "DetachPolicy") || ($.eventName = "DisablePolicyType") || ($.eventName = "EnablePolicyType") || ($.eventName = "InviteAccountToOrganization") || ($.eventName = "LeaveOrganization") || ($.eventName = "MoveAccount") || ($.eventName = "RemoveAccountFromOrganization") || ($.eventName = "UpdatePolicy") || ($.eventName = "UpdateOrganizationalUnit"))}"  4. 3번에서 확인한 MetricName 값을 활용하여 해당 지표 이름으로 설정된 CloudWatch alarms이 있는지 확인한다. # aws cloudwatch describe-alarms --query 'MetricAlarms[?MetricName=='<metric\_name>']'  5. 해당 알람의 AlarmActions 값을 복사한다. \* AlarmActions 값은 <sns\_topic\_arn>이다.  6. SNS 주제에 1명 이상의 활성화된 구독자가 있는지 확인한다. 또한, 최소 1개 이상의 유효한 SubscriptionARN을 가진 구독이 존재해야 한다.  # aws sns list-subscriptions-by-topic --topic-arn <sns\_topic\_arn>  \* SubscriptionARN의 예시는 다음과 같다.  "SubscriptionArn": "arn:aws:sns:<region>:<aws\_account\_number>:<SNS\_topic\_name>:<Subscription\_ID>" | | |
| Reference | 4.15 Ensure a log metric filter and alarm exists for AWS Organizations changes (Automated) | | |
| Site | https://docs.aws.amazon.com/awscloudtrail/latest/userguide/receive-cloudtrail-log-files-from-multiple-regions.html https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html https://docs.aws.amazon.com/sns/latest/dg/welcome.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | SNS 주제 액세스 정책 설정(6.2.1) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | SNS | | |
| 요약 | SNS 주제의 액세스 정책은 특정 소유자 또는 지정된 AWS 계정만 게시 및 구독할 수 있도록 설정하는 것을 권장한다. | | |
| 설명 | 모든 사용자가 메시지를 구독하거나 보낼 수 있도록 SNS(Simple Notification Service) 정책을 구성할 경우 데이터가 유출되거나 과금이 발생할 수 있다. | | |
| 진단 기준 | ▶ 양호 : SNS 주제의 액세스 정책 중 게시하거나 구독할 수 있는 사람이 "주제 소유자만(Only the topic owner)" 또는 "지정된 AWS 계정만(Only the specified AWS accounts)"으로 설정되어 있는 경우    ▶ 취약 : SNS 주제의 액세스 정책 중 게시하거나 구독할 수 있는 사람이 "모든 사람(Everyone)"으로 설정되어 있는 경우   ▶ Default Value : 주제 소유자만 | | |
| 조치 방안 | [AWS Console] 1. https://us-west-2.console.aws.amazon.com/sns/ 를 통해 Amazon SNS에 접속한다. 2. 좌측 사이드 바에서 '주제(Topics)'를 클릭한다.  3. 해당 주제를 클릭한다. 4. 편집(Edit) 버튼을 클릭한다. 5. 액세스 정책(Access policy) 항목을 선택하여 다음 구문을 삭제한다.  ※ 모든 사람이 주제에 메세지를 게시할 수 있도록 설정  {  "Sid": "\_\_console\_pub\_0",  "Effect": "Allow",  "Principal": {  "AWS": "\*"  },  "Action": "SNS:Publish",  "Resource": ""  }   ※ 모든 사람이 이 주제를 구독할 수 있는 사용자로 설정  {  "Sid": "\_\_console\_sub\_0",  "Effect": "Allow",  "Principal": {  "AWS": "\*"  },  "Action": [  "SNS:Subscribe",  "SNS:Receive"  ],  "Resource": ""  }  6. 변경 사항 저장(Save changes) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/sns/ 를 통해 Amazon SNS에 접속한다. 2. 좌측 사이드 바에서 '주제(Topics)'를 클릭한다.  3. 각 주제를 클릭한다. 4. 액세스 정책(Access policy) 탭을 선택한다. 5. 다음 구문이 없는지 확인한다.  ※ 모든 사람이 주제에 메세지를 게시할 수 있도록 설정  {  "Sid": "\_\_console\_pub\_0",  "Effect": "Allow",  "Principal": {  "AWS": "\*"  },  "Action": "SNS:Publish",  "Resource": ""  }   ※ 모든 사람이 이 주제를 구독할 수 있는 사용자로 설정  {  "Sid": "\_\_console\_sub\_0",  "Effect": "Allow",  "Principal": {  "AWS": "\*"  },  "Action": [  "SNS:Subscribe",  "SNS:Receive"  ],  "Resource": ""  }※ | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/sns/latest/dg/sns-access-policy-use-cases.html https://docs.aws.amazon.com/sns/latest/dg/sns-access-policy-use-cases.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | SNS 주제 암호화 설정(6.2.2) | 위험도 | Medium |
| 대분류 | Monitoring | | |
| 중분류 | SNS | | |
| 요약 | SNS 주제에 서버 측 암호화를 활성화하는 것을 권장한다. | | |
| 설명 | Amazon SNS(Simple Notification Service)는 기본적으로 전송 중 암호화를 지원한다. SNS 주제에 암호화를 설정할 경우, 서버 측 암호화(Sever-side encryption)를 사용하여 유휴 시 데이터를 암호화할 수 있다. 서버 측 암호화는 Amazon SNS 주제의 메시지 본문을 암호화하며, 주제 메타데이터(주제 이름 및 속성), 메시지 메타데이터(주제, 메시지 ID, 타임스탬프 및 속성), 주제별 지표는 암호화되지 않는다. SNS가 메시지를 수신하는 즉시 이를 암호화하여 저장하고, 전송된 경우에만 메시지를 해독한다. 민감한 데이터를 보호하기 위하여 유휴 데이터를 암호화하는 것을 권장한다. 권한 없는 사용자가 데이터에 접근하는 것을 제한하고 보안 계층을 강화하기 위해 유휴 데이터에 대한 암호화를 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : SNS 주제의 암호화가 활성화되어 있는 경우   ▶ 취약 : SNS 주제의 암호화가 비활성화되어 있는 경우  ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/sns/ 를 통해 Amazon SNS에 접속한다. 2. 좌측 사이드 바에서 '주제(Topics)'를 클릭한다.  3. 해당 주제를 클릭한다. 4. 암호화(Encryption) 탭을 선택한다. 5. 상단의 '편집(Edit)' 버튼을 클릭한다.  6. 암호화 항목에서 "암호화 활성화(Enable encryption)"를 선택한다.  7. 변경 사항 저장(Save changes) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 SNS 주제에 암호화를 설정한다.  # aws sns set-topic-attributes --topic-arn <topic arn> --attribute-name "KmsMasterKeyId" --attribute-value <key> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/sns/ 를 통해 Amazon SNS에 접속한다. 2. 좌측 사이드 바에서 '주제(Topics)'를 클릭한다.  3. 각 주제를 클릭한다. 4. 암호화(Encryption) 탭을 선택한다. 5. 암호화 항목이 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 SNS 주제에 암호화가 설정되어 있는지 확인한다. "KmsMasterKeyId" 항목이 존재하지 않을 경우, 암호화가 비활성화된 것을 의미한다.  # aws sns get-topic-attributes --topic-arn <topic arn>   예시는 다음과 같다.  {  "Attributes": {  ...  "KmsMasterKeyId": "alias/aws/sns",  ...  } } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/sns/latest/dg/sns-server-side-encryption.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/sns-encrypted-kms.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-sns-1 | | |

**Networking**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 네트워크 ACL에서 인터넷 원격 접속 제한(7.1.1) | 위험도 | High |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | 네트워크 ACL에서 SSH와 RDP 같은 원격 콘솔 서비스의 연결을 제한하는 것을 권장한다. | | |
| 설명 | 네트워크 ACL(Network Access Control List)은 AWS 리소스에 대한 수신 및 송신 네트워크 트래픽의 stateless 필터링을 제공한다. SSH(Secure Shell), RDP(Remote Desktop Protocol) 와 같은 원격 콘솔 서비스의 연결을 제한할 경우, 리소스의 손상 위험을 줄일 수 있기 때문에 네트워크 ACL 규칙을 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 네트워크 ACL에 "소스(Source) : 0.0.0.0/0, 프로토콜(Protocol) : TCP, 포트 범위(Port range) : 22 또는 3389, 허용/거부(Allow/Deny) : Allow" 인 규칙이 포함되어 있지 않은 경우   ▶ 취약 : 각 네트워크 ACL에 "소스 : 0.0.0.0/0, 프로토콜 : TCP, 포트 범위 : 22 또는 3389, 허용/거부 : Allow" 인 규칙이 포함되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '네트워크 ACL(Network ACLs)'를 클릭한다.  3. 해당 네트워크 ACL ID를 클릭한다. 4. 인바운드 규칙(Inbound rules) 탭을 클릭한다. 5. '인바운드 규칙 편집(Edit inbound rules)' 버튼을 클릭한다. 6. 해당 규칙 우측의 '제거(Remove)' 버튼을 클릭한다. 7. 변경 사항 저장(Save changes)을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '네트워크 ACL(Network ACLs)'를 클릭한다.  3. 각 네트워크 ACL ID를 클릭한다. 4. 인바운드 규칙(Inbound rules) 탭을 클릭한다. 5. 아래와 같은 규칙이 존재하는지 확인한다.   - 소스(Source) : 0.0.0.0/0  - 프로토콜(Protocol) : TCP   - 포트 범위(Port range) : 22  - 허용/거부(Allow/Deny) : Allow   - 소스(Source) : 0.0.0.0/0  - 프로토콜(Protocol) : TCP  - 포트 범위(Port range) : 3389  - 허용/거부(Allow/Deny) : Allow  ※ 포트 범위가 "ALL" 또는 "0-1024"와 같이 22와 3389를 포함하는 경우에도 외부에서 접근할 수 없도록 설정해야 한다. | | |
| Reference | 5.1 Ensure no Network ACLs allow ingress from 0.0.0.0/0 to remote server administration ports (Automated) | | |
| Site | https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html https://docs.aws.amazon.com/vpc/latest/userguide/VPC\_Security.html#VPC\_Security\_Comparison | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 인스턴스 보안 그룹에서 인터넷 원격 접속 제한(7.1.2) | 위험도 | High |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | 보안 그룹에서 SSH와 RDP 같은 원격 콘솔 서비스의 연결을 제한하는 것을 권장한다. | | |
| 설명 | 보안 그룹(Security Group)은 AWS 리소스에 대한 수신 및 송신 네트워크 트래픽의 stateful 필터링을 제공한다. SSH(Secure Shell), RDP(Remote Desktop Protocol) 와 같은 원격 콘솔 서비스의 연결을 제한할 경우, 리소스의 손상 위험을 줄일 수 있기 때문에 보안 규칙을 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 보안 그룹에 "소스(Source) : 0.0.0.0/0, 프로토콜(Protocol) : TCP, 포트 범위(Port range) : 22 또는 3389" 인 규칙이 포함되어 있지 않은 경우   ▶ 취약 : 각 보안 그룹에 "소스 : 0.0.0.0/0, 프로토콜 : TCP, 포트 범위 : 22 또는 3389" 인 규칙이 포함되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '보안 그룹(Security Groups)'을 클릭한다.  3. 해당 보안 그룹 ID를 클릭한다. 4. 인바운드 규칙(Inbound rules) 탭을 클릭한다. 5. '인바운드 규칙 편집(Edit inbound rules)' 버튼을 클릭한다. 6. 해당 규칙 우측의 '삭제(Delete)' 버튼을 클릭한다. 7. 규칙 저장(Save rules)을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '보안 그룹(Security Groups)'을 클릭한다.  3. 각 보안 그룹 ID를 클릭한다. 4. 인바운드 규칙(Inbound rules) 탭을 클릭한다. 5. 아래와 같은 규칙이 존재하는지 확인한다.   - 소스(Source) : 0.0.0.0/0  - 프로토콜(Protocol) : TCP   - 포트 범위(Port range) : 22   - 소스(Source) : 0.0.0.0/0  - 프로토콜(Protocol) : TCP  - 포트 범위(Port range) : 3389  ※ 포트 범위가 "ALL" 또는 "0-1024"와 같이 22와 3389를 포함하는 경우에도 외부에서 접근할 수 없도록 설정해야 한다. | | |
| Reference | 5.2 Ensure no security groups allow ingress from 0.0.0.0/0 to remote server administration ports (Automated) | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 기본 보안 그룹 사용 제한(7.1.3) | 위험도 | High |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | VPC의 default 보안 그룹에서 모든 트래픽을 제한하도록 설정하는 것을 권장한다. | | |
| 설명 | 보안 그룹(Security Group)은 AWS 리소스에 대한 수신 및 송신 네트워크 트래픽의 stateful 필터링을 제공한다. VPC에는 기본적으로 ‘default 보안 그룹’이 존재하며 모든 인바운드 및 아웃바운드 트래픽을 허용하는 보안 규칙이 설정되어 있다. 해당 규칙은 보안 그룹에 할당된 인스턴스 간의 모든 트래픽을 허용한다. 인스턴스를 시작할 때 보안 그룹을 지정하지 않을 경우, default 보안 그룹이 인스턴스에 자동으로 할당된다. 리소스의 노출과 손상 위험을 줄이기 위해 기본 보안 그룹에서 모든 트래픽을 제한하는 것을 권장한다.  ※ default 보안 그룹은 삭제할 수 없다. | | |
| 진단 기준 | ▶ 양호 : 각 VPC의 default 보안 그룹에 모든 인바운드, 아웃바운드 규칙이 존재하지 않는 경우   ▶ 취약 : 각 VPC의 default 보안 그룹에 인바운드, 아웃바운드 규칙이 존재하는 경우   ▶ Default Value : 모든 트래픽 허용(인바운드, 아웃바운드 규칙 존재) | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '보안 그룹(Security Groups)'을 클릭한다.  3. 보안 그룹 이름이 "default" 로 설정되어 있는 각 보안 그룹 ID를 클릭한다. 4. 인바운드 규칙(Inbound rules) 탭을 클릭한다. 5. '인바운드 규칙 편집(Edit inbound rules)' 버튼을 클릭한다. 6. 모든 규칙의 '삭제(Delete)' 를 클릭한다. 7. 규칙 저장(Save rules)을 클릭한다. 8. 아웃바운드 규칙(Outbound rules) 탭을 클릭한다. 9. '아웃바운드 규칙 편집(Edit outbound rules)' 버튼을 클릭한다. 10. 모든 규칙의 '삭제(Delete)'를 클릭한다. 11. 규칙 저장(Save rules)을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '보안 그룹(Security Groups)'을 클릭한다.  3. 보안 그룹 이름이 "default" 로 설정되어 있는 각 보안 그룹 ID를 클릭한다. 4. 인바운드 규칙(Inbound rules) 탭을 클릭한다. 5. 인바운드 규칙이 없는지 확인한다. 6. 아웃바운드 규칙(Outbound rules) 탭을 클릭한다. 7. 아웃바운드 규칙이 없는지 확인한다. | | |
| Reference | 5.3 Ensure the default security group of every VPC restricts all traffic (Automated) | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/default-custom-security-groups.html#default-security-group https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/vpc-default-security-group-closed.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-ec2-2 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | VPC 내 멀티 서브넷 구성(7.1.6) | 위험도 | Low |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | VPC 내 서브넷을 여러 개로 설정하여 계층화 된 아키텍처를 구성하는 것을 권장한다. | | |
| 설명 | VPC 내 계층화된 아키텍처를 구성할 경우 악의적인 사용자에 의해 손상되는 피해 범위를 줄일 수 있다. 단일 네트워크 구성시 모든 범위가 손상되지만 계층화된 경우 피해 범위가 특정 범위로 제한되어 피해를 줄이는 데 도움이 된다. | | |
| 진단 기준 | ▶ 양호 : 각 VPC 내 서브넷이 2개 이상으로 설정되어 있는 경우   ▶ 취약 : 각 VPC 내 서브넷이 단일로 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '서브넷(Subnets)'을 클릭한다.  3. 서브넷 생성(Create subnet) 버튼을 클릭한다. 4. 단일 서브넷으로 구성된 VPC를 선택한다. 5. 서브넷 이름, 가용 영역, IPv4 CIDR 블록 등을 설정한 후, 서브넷 생성(Create subnet) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '서브넷(Subnets)'을 클릭한다.  3. 각 VPC 내 서브넷이 2개 이상인지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/vpc/latest/userguide/VPC\_SecurityGroups.html https://docs.aws.amazon.com/vpc/latest/userguide/VPC\_Scenario2.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 인스턴스 보안 그룹에서 80/443 포트 외 인터넷 노출 제한(7.1.7) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | 인스턴스의 보안 그룹 내 80, 443 포트 외의 인바운드 규칙은 특정 IP에 한하여 허용하도록 설정하는 것을 권장한다. | | |
| 설명 | 일반적으로 운영하는 서버의 경우, 인터넷에서 80, 443 포트만 접근 가능하도록 설정하는 것을 권장한다. 업무상 다른 포트를 허용하도록 설정해야 하는 경우에는 특정 IP에 한하여 접근 가능하도록 설정하거나 보안 그룹을 이용하여 특정 IP에 한하여 접근 가능하도록 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 운영 중인 인스턴스의 보안 그룹 인바운드 규칙 중 80/443 포트 외 특정 IP에서 특정 포트로 설정되어 있는 경우   ▶ 취약 : 운영 중인 인스턴스의 보안 그룹 인바운드 규칙 중 80/443 포트 외 0.0.0.0/0에서 특정 포트로 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 클릭한다.  4. 보안(Security) 탭을 클릭한다. 5. 보안 그룹(Security Groups)의 ID를 클릭한다.  6. '인바운드 규칙 편집(Edit inbound rules)' 버튼을 클릭한다. 7. 아래와 같은 규칙을 제외한 다른 규칙은 모두 '삭제(Delete)' 버튼을 클릭하여 삭제하거나 특정 IP만 접근할 수 있도록 설정한다.   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 80   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 443  8. 규칙 저장(Save rules)을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 보안(Security) 탭을 클릭한다. 5. 인바운드 규칙(Inbound rules)에서 아래와 같은 규칙을 제외한 다른 규칙이 존재하는지 확인한다.   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 80   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 443 | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 네트워크 ACL에서 80/443 포트 외 인터넷 노출 제한(7.1.8) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | 인스턴스의 서브넷 내 80, 443 포트 외의 네트워크 ACL 인바운드 규칙은 특정 IP에 한하여 허용하도록 설정하는 것을 권장한다. | | |
| 설명 | 일반적으로 운영하는 서버의 경우, 인터넷에서 80, 443 포트만 접근 가능하도록 설정하는 것을 권장한다. 업무상 다른 포트를 허용하도록 설정해야 하는 경우에는 특정 IP에 한하여 접근 가능하도록 설정하거나 네트워크 ACL(Network Access Control List)을 이용하여 특정 IP에 한하여 접근 가능하도록 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 운영 중인 인스턴스의 서브넷 NACL 인바운드 규칙 중 80/443 포트 외 특정 IP에서 특정 포트로 접근이 허용되도록 설정되어 있는 경우   ▶ 취약 : 운영 중인 인스턴스의 서브넷 NACL 인바운드 규칙 중 80/443 포트 외 0.0.0.0/0에서 특정 포트로 접근이 허용되도록 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 해당 인스턴스를 클릭한다.  4. 인스턴스 요약(Instance summary)에서 '서브넷 ID(Subnet ID)'를 클릭한다. 5. 서브넷 ID를 클릭한다.  6. 네트워크 ACL(Network ACL) 탭을 선택한다.  7. '네트워크 ACL 연결 편집(Edit network ACL association)' 버튼을 클릭하여, '현재 네트워크 ACL(Current network ACL)' ID를 클릭한다.  8. 네트워크 ACL ID를 클릭한다.  9. 인바운드 규칙(Inbound rules) 탭을 선택하여, '인바운드 규칙 편집(Edit inbound rules)' 버튼을 클릭한다. 10. 아래와 같은 규칙을 제외한 다른 규칙은 모두 '제거(Remove)' 버튼을 클릭하여 삭제하거나 특정 IP만 접근할 수 있도록 설정한다.   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 80  - 허용/거부(Allow/Deny) : Allow   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 443  - 허용/거부(Allow/Deny) : Allow  11. 변경 사항 저장(Save changes)을 클릭한다. ※ 기존 네트워크 ACL에서 변경할 경우 다른 서비스에도 동일하게 적용해도 되는 규칙인지 검토 후에 적용하는 것을 권장한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '인스턴스(Instances)'를 클릭한다.  3. 각 인스턴스를 클릭한다.  4. 인스턴스 요약(Instance summary)에서 '서브넷 ID(Subnet ID)'를 클릭한다. 5. 서브넷 ID를 클릭한다.  6. 네트워크 ACL(Network ACL) 탭을 클릭한다. 7. 인바운드 규칙(Inbound rules) 에서 아래와 같은 규칙을 제외한 다른 규칙이 존재하는지 확인한다.   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 80  - 허용/거부(Allow/Deny) : Allow   - 소스(Source) : 0.0.0.0/0  - 포트 범위(Port range) : 443  - 허용/거부(Allow/Deny) : Allow | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html https://docs.aws.amazon.com/vpc/latest/userguide/VPC\_Security.html#VPC\_Security\_Comparison | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | VPC 플로우 로그 활성화(7.1.9) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | VPC의 "거부(reject)" 패킷에 대해 VPC 플로우 로그를 활성화하는 것을 권장한다. | | |
| 설명 | VPC 플로우 로그를 통해 VPC에서 네트워크 인터페이스에 들어오고 나가는 IP 트래픽에 대한 정보를 캡쳐할 수 있으며, 비정상적인 트래픽을 탐지할 수 있다. 흐름 로그를 생성한 후 Amazon CloudWatch 로그에서 해당 데이터를 보거나 검색할 수 있다. 기본적으로 CloudWatch 로그는 로그 그룹에 대해 특정 보존 기간이 정의되어 있지 않은 로그는 무기한으로 저장한다. 침해사고 발생 시 이를 인지하는데 평균적으로 210일 정도가 소요되기 때문에, 침해사고 조사 및 대응을 위해 최소 1년 이상 로그를 보존하도록 보존 정책을 설정해야 한다. 또한, 기간이 지난 로그는 삭제하지 않고 저렴한 스토리지 서비스에 장기간 보관하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 VPC의 VPC 플로우 로그(Flow logs)가 생성되어 있는 경우   ▶ 취약 : 각 VPC의 VPC 플로우 로그가 생성되어 있지 않은 경우   ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc 를 통해 VPC 대시보드에 접속한다. 2. 좌측 사이드 바에서 '가상 프라이빗 클라우드(VIRTUAL PRIVATE CLOUD)'의 'VPC(Your VPCs)'를 클릭한다.  3. 해당 VPC를 클릭한다.  4. 플로우 로그(Flow logs) 탭을 클릭한다. 5. 흐름 로그 생성(Create flow log) 버튼을 클릭한다. 6. 플로우 로그 설정(Flow log settings) 중 필터(Filter) 항목에서 "거부(Reject)"를 선택한다. 7. 대상 로그 그룹(Destination log group)과 IAM 역할(IAM role)을 선택한다. 8. 흐름 로그 생성(Create flow log) 버튼을 클릭한다. ※ 필터를 "거부"로 설정할 경우, 로깅 데이터 누적량이 현저히 줄어들며 위반 탐지 및 문제 해결을 위한 정보를 제공한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc 를 통해 VPC 대시보드에 접속한다. 2. 좌측 사이드 바에서 가상 프라이빗 클라우드(VIRTUAL PRIVATE CLOUD)의 'VPC(Your VPCs)'를 클릭한다.  3. 각 VPC를 클릭한다.  4. 플로우 로그(Flow logs) 탭을 클릭한다. 5. 플로우 로그의 '상태(Status)'가 "활성(Active)" 인지 확인한다. | | |
| Reference | 3.9 Ensure VPC flow logging is enabled in all VPCs (Automated) | | |
| Site | https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/WhatIsCloudWatchLogs.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/vpc-flow-logs-enabled.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 사이트 간 VPN 연결의 터널 작동 상태 설정(7.1.10) | 위험도 | Low |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | VPC의 가용성을 위해 사이트 간 VPN 연결에 두 개의 터널을 구성하고, 작동 중인 상태로 유지하는 것을 권장한다. | | |
| 설명 | 사이트 간 VPN 연결(Site-to-Site VPN)의 모든 VPN 터널의 상태가 "UP" 인지 확인하는 것을 권장한다. 사이트 간 VPN 연결을 생성할 경우, VPC에서 온프레미스 네트워크에 대한 액세스를 활성화할 수 있다. VPN 터널은 데이터가 고객 네트워크에서 AWS와 주고받을 수 있는 암호화된 링크이며, 각 사이트 간 VPN 연결에는 고가용성을 위해 동시에 사용할 수 있는 두 개의 VPN 터널이 포함되어 있다. AWS 내 디바이스에 장애가 발생할 경우, VPN 연결은 두번째 터널로 자동으로 장애 조치되어 액세스가 중단되지 않는다. | | |
| 진단 기준 | ▶ 양호 : 각 사이트 간 VPN 연결 내 모든 터널(Tunnel 1, Tunnel 2)의 상태가 "UP" 인 경우   ▶ 취약 : 각 사이트 간 VPN 연결 내 일부 터널의 상태가 "DOWN" 인 경우   ▶ Default Value : UP | | |
| 조치 방안 | - | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '사이트 간 VPN 연결(Site-to-Site VPN Connections)'을 클릭한다. 3. 각 VPN 연결을 클릭한다.  4. 터널 세부 정보(Tunnel Details) 탭을 선택한다.  5. Tunnel 1, Tunnel 2의 '상태(Status)'가 "UP" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 사이트 간 VPN 연결의 두 터널의 상태가 "UP" 으로 설정되어 있는지 확인한다.  # aws ec2 describe-vpn-connections --vpn-connection-ids <vpn connection id>  예시는 다음과 같다.  "VgwTelemetry": [  {  "AcceptedRouteCount": 0,  "LastStatusChange": "2020-07-29T10:35:11.000Z",  "OutsideIpAddress": "203.0.113.3",  "Status": "UP",  "StatusMessage": ""  },  {  "AcceptedRouteCount": 0,  "LastStatusChange": "2020-09-02T09:09:33.000Z",  "OutsideIpAddress": "203.0.113.5",  "Status": "UP",  "StatusMessage": ""  }  ] | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/vpn/latest/s2svpn/VPNTunnels.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/vpc-vpn-2-tunnels-up.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | NAT 게이트웨이 관리(7.1.11) | 위험도 | High |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | 프라이빗 서브넷 내 EC2 인스턴스가 패치, 소프트웨어 업데이트 등을 위해 인터넷 연결이 필요할 경우 NAT 게이트웨이를 사용하는 것을 권장한다. | | |
| 설명 | NAT(Network Address Translation) 게이트웨이는 프라이빗 서브넷에 존재하는 EC2 인스턴스가 인터넷과 직접적인 연결(퍼블릭 IP 또는 인터넷 게이트웨이) 없이도 패치나 소프트웨어 업데이트 등을 다운로드 할 수 있도록 하는 관리형 서비스이다. NAT 게이트웨이는 외부로 연결을 요청하는 EC2 인스턴스의 프라이빗 IP를 NAT 게이트웨이의 퍼블릭 IP와 공유하여 외부로 전달하고, 수신되는 트래픽을 요청한 내부의 EC2 인스턴스로 돌려준다. 5Gbps의 대역폭을 지원하고 사용자의 트래픽에 따라 최대 45Gpbs까지 자동확장이 가능하며, 자체적으로 중복적으로 구현되어 있어 고가용성을 유지할 수 있다.  ※ NAT 게이트웨이를 활성화할 경우, NAT 게이트웨이 생성 및 사용에 대한 요금, EC2 인스턴스 데이터 전송 요금 등이 부과될 수 있다. ※ NAT 게이트웨이는 일부 AWS 리전에서 사용이 불가능하다. | | |
| 진단 기준 | ▶ 양호 : 퍼블릭 NAT 게이트웨이가 존재하지 않는 경우   ▶ 취약 : 퍼블릭 NAT 게이트웨이가 존재하는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 'NAT 게이트웨이(NAT gateways)'를 클릭한다. 3. 불필요한 퍼블릭 NAT 게이트웨이를 클릭한다.  4. 오른쪽 상단의 '삭제(Delete)' 버튼을 클릭한다.  5. 삭제(Delete)를 입력한다.   [AWS CLI] 1. 다음 명령어를 실행하여 퍼블릭 NAT 게이트웨이를 삭제한다.  # aws ec2 delete-nat-gateway --nat-gateway-id <nat gateway id> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 'NAT 게이트웨이(NAT gateways)'를 클릭한다. 3. 불필요한 퍼블릭 NAT 게이트웨이가 존재하는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 불필요한 퍼블릭 NAT 게이트웨이가 존재하는지 확인한다. 퍼블릭 NAT 게이트웨이의 경우, "PublicIP"를 가지고 있다.  # aws ec2 describe-nat-gateways | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 인터넷 게이트웨이 관리(7.1.12) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | VPC | | |
| 요약 | 직접적인 인터넷 접근이 필요한 경우에만 VPC에 인터넷 게이트웨이를 연결하는 것을 권장한다. | | |
| 설명 | 격리된 네트워크 환경인 VPC 내 리소스는 기본적으로 인터넷을 사용할 수 없다. VPC와 인터넷 간 통신을 위해서는 인터넷 게이트웨이가 필요하다. 인터넷 게이트웨이가 연결된 퍼블릭 서브넷에는 직접적인 인터넷 액세스가 필요한 인스턴스만을 배치하는 것을 권장한다. 직접적인 인터넷 연결이 필요하지 않은 경우 프라이빗 서브넷을 사용하고, 프라이빗 서브넷 내 인스턴스가 패치, 소프트웨어 업데이트 등을 위해 인터넷에 연결해야 하는 경우 배스천 호스트 또는 NAT 게이트웨이를 사용하는 것을 권장한다. 또한, 필요한 최소 네트워크 경로로만 VPC 서브넷 라우팅 테이블을 구성해야 한다.  ※ 인터넷을 사용하고자 하는 리소스는 퍼블릭 IP 또는 탄력적 IP를 가지고 있어야 한다.  ※ 인터넷 게이트웨이는 추가 요금이 부과되지 않는다. | | |
| 진단 기준 | ▶ 양호 : 생성된 인터넷 게이트웨이가 사용 중(Attached)인 경우   ▶ 취약 : 생성된 인터넷 게이트웨이가 미사용 중(Detached)인 경우   ▶ Default Value : Detached | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '인터넷 게이트웨이(Internet gateways)'를 클릭한다.  3. '상태(State)'가 "Detached" 인 인터넷 게이트웨이를 체크한다.  4. 상단의 '작업(Actions)' 버튼을 클릭하여, "인터넷 게이트웨이 삭제(Delete internet gateway)"를 선택한다.  5. 인터넷 게이트웨이 삭제(Delete internet gateway) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 인터넷 게이트웨이를 삭제한다.  # aws ec2 delete-internet-gateway --internet-gateway-id <igw id> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/vpc/home 를 통해 AWS VPC에 접속한다. 2. 좌측 사이드 바에서 '인터넷 게이트웨이(Internet gateways)'를 클릭한다.  3. 각 인터넷 게이트웨이의 '상태(State)'가 "Attached" 인지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 인터넷 게이트웨이가 VPC에 연결되어 있는지 확인한다.  # aws ec2 describe-internet-gateways  예시는 다음과 같다.  "Attachments": [  {  "State": "available",  "VpcId": "<vpc id>"  } ] | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/vpc/latest/userguide/VPC\_Internet\_Gateway.html https://docs.aws.amazon.com/vpc/latest/userguide/infrastructure-security.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudFront 전송 중 데이터 암호화(7.2.1) | 위험도 | High |
| 대분류 | Networking | | |
| 중분류 | CloudFront | | |
| 요약 | 전송 중 데이터를 암호화하기 위해 최종 사용자가 HTTPS를 사용하여 파일을 요청하도록 CloudFront를 구성하는 것을 권장한다. | | |
| 설명 | Viewer Protocol Policy 항목을 "Redirect HTTP to HTTPS" 또는 "HTTPS Only" 를 설정 시, CloudFront 엣지 노드와 백엔드 리소스 간 전송되는 트래픽이 암호화되어 전송된다. 또한 웹 기반 Origin은 최신 프로토콜을 사용해야 한다. 민감 데이터의 암호화를 추가적으로 원할 경우, 필드 레벨 암호화를 사용하는 것을 권장한다. HTTPS를 사용할 경우, CloudFront에서 최종 사용자 또는 Origin과 통신할 때 전송 중 데이터 암호화하여 중간자 공격을 방어할 수 있다.  ※ HTTP and HTTPS 설정 시 allow-all로 표시된다. | | |
| 진단 기준 | ▶ 양호 : Viewer Protocol Policy 항목이 "Redirect HTTP to HTTPS" 또는 "HTTPS Only"로 설정되어 있는 경우   ▶ 취약 : Viewer Protocol Policy 항목이 "HTTP and HTTPS"로 설정되어 있는 경우   ▶ Default Value : HTTP and HTTPS | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudfront 를 통해 AWS CloudFront에 접속한다. 2. 해당 Distribution ID를 클릭한다.  3. Behaviors 탭을 클릭한다.  4. 해당 Behaviors를 선택한 후, 상단의 'Edit'을 클릭한다. 5. 'Viewer Protocol Policy' 항목을 "Redirect HTTP to HTTPS" 또는 "HTTPS Only"로 설정한다.  6. 하단의 'Yes, Edit' 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudfront 를 통해 AWS CloudFront에 접속한다. 2. 각 Distribution ID를 클릭한다.  3. Behaviors 탭을 클릭한다.  4. 'Viewer Protocol Policy' 항목이 "Redirect HTTP to HTTPS" 또는 "HTTPS Only" 인지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/using-https.html https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/using-https-viewers-to-cloudfront.html https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/field-level-encryption.html https://docs.aws.amazon.com/cloudfront/latest/APIReference/API\_CacheBehavior.html  https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/cloudfront-viewer-policy-https.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-cloudfront-3 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudFront OAI 설정(7.2.3) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | CloudFront | | |
| 요약 | Amazon S3 버킷을 CloudFront 배포의 Origin으로 사용할 경우, OAI를 설정하여 S3의 URL을 통해 우회 접근하는 것을 방지해야 한다. | | |
| 설명 | OAI(Origin Access Identity)를 사용하여 사용자의 액세스를 제한할 경우, 다른 사용자가 Amazon S3 URL을 통해 우회 접근하는 것을 방지할 수 있다. CloudFront URL을 통해서만 S3 내 객체에 액세스할 수 있도록 OAI를 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : OAI(Origin Access Identity)가 생성되어 있고 Restrict Bucket Access 항목이 "Yes"로 설정되어 있는 경우   ▶ 취약 : OAI가 존재하지 않거나 Restrict Bucket Access 항목이 "No"로 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudfront 를 통해 AWS CloudFront에 접속한다. 2. 해당 Distribution ID를 클릭한다.  3. Origins and Origin Groups 탭을 클릭한다. 4. 'Create Orgin' 버튼을 클릭한다. 5. 각 항목을 설정한다.  ※ 'Restrict Bucket Access' 항목은 "Yes"로 설정한다. ※ 'Grant Read Permissions on Bucket '항목은 직접 정책을 설정할 경우 "No, I Will Update Permissions"를 선택하고, 자동으로 권한을 부여하고자 하는 경우에는 "Yes, Update Bucket Policy"를 선택한다. 사용자가 기존 S3 URL를 사용하여 버킷의 파일에 액세스할 수 있는 권한이 있는 경우 해당 권한을 필수적으로 삭제해야 한다. 6. 하단의 Create 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudfront 를 통해 AWS CloudFront에 접속한다. 2. 각 Distribution ID를 클릭한다.  3. Origins and Origin Groups 탭을 클릭한다. 4. Orgins 가 존재하는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/data-protection-summary.html https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-restricting-access-to-s3.html https://aws.amazon.com/ko/premiumsupport/knowledge-center/s3-website-cloudfront-error-403/ https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/cloudfront-origin-access-identity-enabled.html https://docs.aws.amazon.com/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-cloudfront-2 https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-restricting-access-to-s3.html#private-content-creating-oai | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CloudFront 로깅 활성화(7.2.4) | 위험도 | Low |
| 대분류 | Networking | | |
| 중분류 | CloudFront | | |
| 요약 | 보안 및 액세스 감사를 위해 CloudFront 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | 로깅을 활성화하기 위해서는 버킷에 대한 ACL을 업데이트 해야 하며, awslogdelivery 계정에 FULL\_CONTROL 권한을 부여해야 한다. 액세스 로그 정보는 ClondFront에서 수신하는 모든 사용자 요청에 대한 세부 정보를 포함하며, 이는 보안 및 액세스 감사에 유용하므로 로깅을 활성화하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 CloudFront Distribution에 'Standard Logging' 항목이 "On"으로 설정되어 있는 경우   ▶ 취약 : 각 CloudFront Distribution에 'Standard Logging 항목이 "Off"로 설정되어 있지 않은 경우   ▶ Default Value : Off | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudfront 를 통해 AWS CloudFront에 접속한다. 2. 해당 Distribution ID를 클릭한다.  3. General 탭을 클릭한다.  4. Settings 우측의 'Edit' 버튼을 클릭한다.  5. "Standartd logging" 항목을 'On'으로 선택한 후, S3 버킷을 설정한다.  6. Save changes 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/cloudfront 를 통해 AWS CloudFront에 접속한다.  2. 각 Distribution ID를 클릭한다.  3. General 탭을 클릭한다.  4. "Standard logging" 항목이 "On"으로 설정되어 있는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/logging-and-monitoring.html https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/AccessLogs.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ELB 보안 정책 최신 버전 설정(7.3.1) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | ELB의 보안 정책 설정 시 AWS에서 사전 정의된 보안 정책 중 최신 버전을 설정하는 것을 권장한다. | | |
| 설명 | AWS에서 사전 정의된 보안 정책을 사용하는 것을 권장한다. 사용자는 주기적으로 ELB(Elastic Load Balancer)의 보안 정책 업데이트를 확인하여 ELB의 보안 정책을 최신화해야 한다.  ※ 2021-02 기준 - Classic Load Balancer :  ELBSecurityPolicy-2016-08,  ELBSecurityPolicy-TLS-1-2-2017-01,  ELBSecurityPolicy-TLS-1-1-2017-01 인 경우 양호이다.  사용자 지정 보안 정책 시   SSL 프로토콜이 Protocol-TLSv1.1 이상   SSL Ciphers가 DHC-RSA-AES128-SHA, DHE-DSS-AES128-SHA, DES-CBC3-SHA가 포함되어 있지 않을 경우 양호이다.  - Application Load Balancer, Network Load Balancer :  ELBSecurityPolicy-FS-1-2-Res-2020-10,  ELBSecurityPolicy-FS-1-2-Res-2019-08,  ELBSecurityPolicy-FS-1-2-2019-10,  ELBSecurityPolicy-TLS-1-2-Ext-2018-06  ELBSecurityPolicy-TLS-1-2-2017-01 인 경우 양호이다. | | |
| 진단 기준 | ▶ 양호 : ELB의 보안 정책(Security Policy)이 최신 버전으로 설정되어 있는 경우   ▶ 취약 : ELB의 보안 정책이 최신 버전으로 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 로드 밸런싱(Load Balancing)의 '로드밸런서(Load Balancers)'를 클릭한다.  3. 해당 로드밸런서를 선택한다.  4. 리스너(Listeners) 탭을 클릭한다.  5. 리스너 ID가 "HTTPS : 443" 인 리스너를 체크한 후, '편집(Edit)' 버튼을 클릭한다.  6. "보안 정책(Security policy)" 항목에서 최신 보안 정책을 선택한다.  7. 상단의 '업데이트(Update)' 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 로드 밸런싱(Load Balancing)의 '로드밸런서(Load Balancers)'를 클릭한다.  3. 각 로드밸런서를 선택한다.  4. 리스너(Listeners) 탭을 클릭한다.  5. 리스너 ID가 "HTTPS : 443" 인 리스너의 "보안 정책(Security policy)"이 최신 버전으로 설정되어 있는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticloadbalancing/latest/application/create-https-listener.html https://docs.aws.amazon.com/elasticloadbalancing/latest/classic/elb-security-policy-table.html https://docs.aws.amazon.com/cli/latest/reference/elb/describe-load-balancer-policies.html https://docs.aws.amazon.com/elasticloadbalancing/latest/application/listener-update-certificates.html#update-security-policy | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ELB에 사용된 인증서 갱신(7.3.2) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 인증서가 만료되기 30일 이전에 새로운 인증서로 교체하는 것을 권장한다. | | |
| 설명 | ACM(AWS Certificate Manager)에서 발급한 인증서를 로드 밸런서에 배포할 경우 해당 인증서를 자동으로 갱신할 수 있다. ACM은 인증서가 만료되기 전에 인증서를 갱신한다. ACM에 인증서를 가져온 경우 인증서의 만료 날짜를 반드시 모니터링하여 만료되기 30일 전에 인증서를 갱신하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 인증서 만료일이 현재 날짜로부터 30일 이상 남은 경우   ▶ 취약 : 인증서 만료일이 현재 날짜로부터 30일 이내 남은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 로드 밸런싱(Load Balancing)의 '로드밸런서(Load Balancers)'를 클릭한다.  3. 해당 로드밸런서를 선택한다.  4. 리스너(Listeners) 탭을 클릭한다.  5. 해당 리스너의 '인증서 보기/편집(View/edit certificates)' 버튼을 클릭한다.  6. 상단의 '+' 버튼을 선택하여, '인증서 가져오기(Import certificate)' 또는 'ACM'을 클릭한다.  7. '인증서 가져오기'를 선택한 경우, "ACM" 또는 "IAM"을 선택한 후 각 항목을 설정한다. 'ACM'을 선택한 경우, AWS Certificate Manager에서 인증서를 갱신하거나 교체한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 로드 밸런싱(Load Balancing)의 '로드밸런서(Load Balancers)'를 클릭한다.  3. 각 로드밸런서를 선택한다.  4. 리스너(Listeners) 탭을 클릭한다.  5. 각 리스너의 '인증서 보기/편집(View/edit certificates)' 버튼을 클릭한다.  6. 해당 인증서의 만료일(Expires)이 30일 이상 남았는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/elasticloadbalancing/latest/classic/elb-update-ssl-cert.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CLB 교차 영역 로드 밸런싱 활성화(7.3.3) | 위험도 | Low |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 클래식 로드 밸런서에 교차 영역 로드 밸런싱을 활성화하는 것을 권장한다. | | |
| 설명 | CLB(Classic Load Balancer)에 교차 영역 로드 밸런싱을 활성화할 경우, CLB의 각 로드 밸런서 노드가 활성화된 모든 가용 영역의 등록된 인스턴스 간의 요청을 균등하게 분산한다. 때문에 각 가용 영역에서 동일한 수의 인스턴스를 유지할 필요가 줄어들며, 인스턴스 손실을 처리할 수 있는 애플리케이션의 기능이 향상된다. 만약 교차 영역 로드 밸런싱을 비활성화할 경우, 각 로드 밸런서 노드가 해당 가용 영역에 있는 등록된 대상 간에만 트래픽을 분산한다. | | |
| 진단 기준 | ▶ 양호 : 각 CLB에 "교차 영역 로드 밸런싱(Cross-zone load balancing)"이 활성화되어 있는 경우   ▶ 취약 : 각 CLB에 "교차 영역 로드 밸런싱"이 비활성화되어 있는 경우   ▶ Default Value : Enabled ※ AWS Console로 생성할 경우 기본적으로 교차 영역 로드 밸런싱이 활성화되며, API 또는 CLI로 생성할 경우 비활성화된다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 해당 CLB를 클릭한다.  4. 설명(Description) 탭을 선택한다.  5. 속성(Attributes)에서 "교차 영역 로드 밸런싱 설정 변경(Change cross-zone load balancing setting)" 버튼을 클릭한다.  6. "활성화(Enable)"를 선택한 후, 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 CLB의 교차 영역 로드 밸런싱을 활성화한다.  # aws elb modify-load-balancer-attributes --load-balancer-name <name> --load-balancer-attributes CrossZoneLoadBalancing={Enabled=true} | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 CLB를 클릭한다.  4. 설명(Description) 탭을 선택한다.  5. 속성(Attributes)에서 "교차 영역 로드 밸런싱(Cross-zone load balancing)" 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 CLB의 교차 영역 로드 밸런싱이 활성화되어 있는지 확인한다.  # aws elb describe-load-balancer-attributes --load-balancer-name <name>  예시는 다음과 같다.  {  "LoadBalancerAttributes": {  "CrossZoneLoadBalancing": {  "Enabled": true  }  } } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticloadbalancing/latest/userguide/how-elastic-load-balancing-works.html#cross-zone-load-balancing https://docs.aws.amazon.com/elasticloadbalancing/latest/classic/enable-disable-crosszone-lb.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/elb-cross-zone-load-balancing-enabled.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ELB 삭제 방지 활성화(7.3.4) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 로드 밸런서의 삭제 방지 기능을 활성화하는 것을 권장한다. | | |
| 설명 | 삭제 방지 기능을 활성화할 경우, 로드 밸런서가 실수 또는 악의적으로 삭제되는 것을 방지하여 애플리케이션의 가용성을 보장할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 로드 밸런서(ALB, NLB, GWLB)에 "삭제 방지(Deletion protection)"가 활성화되어 있는 경우   ▶ 취약 : 각 로드 밸런서(ALB, NLB, GWLB)에 "삭제 방지"가 비활성화되어 있는 경우   ▶ Default Value : Disabled ※ 해당 항목은 CLB에 해당되지 않는다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 해당 로드 밸런서(ALB, NLB, GWLB)를 클릭한다.  4. 상단의 '작업(Actions)' 버튼을 클릭하여, '속성 편집(Edit attributes)'을 선택한다.  5. "삭제 방지(Deletion protection)" 항목의 "활성화(Enable)"를 선택한다.  6. 저장(Save) 버튼을 클릭한다.   [AWS CLI] ※ ALB, NLB, GWLB의 경우  1. 다음 명령어를 실행하여 해당 로드 밸런서의 삭제 방지를 활성화한다. # aws elbv2 modify-load-balancer-attributes --load-balancer-arn <arn> --attributes Key=deletion\_protection.enabled,Value=true | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 로드 밸런서(ALB, NLB, GWLB)를 클릭한다.  4. 설명(Description) 탭을 선택한다.  5. 속성(Attributes) 에서 "삭제 방지(Deletion protection)" 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] ※ ALB, NLB, GWLB의 경우 1. 다음 명령어를 실행하여 각 로드 밸런서에 삭제 방지가 활성화되어 있는지 확인한다.  # aws elbv2 describe-load-balancer-attributes --load-balancer-arn <arn>  예시는 다음과 같다.  {  "Key": "deletion\_protection.enabled",  "Value": "true"  } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticloadbalancing/latest/application/application-load-balancers.html#deletion-protection https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/elb-deletion-protection-enabled.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-elb-6 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ELB 액세스 로깅 활성화(7.3.5) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 로드 밸런서의 액세스 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | Elastic Load Balancing은 로드 밸런서에 전송된 요청에 대한 세부 정보를 캡처하는 액세스 로그를 제공한다. 로그에는 요청받은 시간, 클라이언트 IP주소, 지연 시간, 요청 경로, 서버 응답과 같은 정보가 포함된다. 액세스 로깅을 활성화할 경우, Elastic Load Balancing가 로그를 캡처하여 지정한 Amazon S3 버킷에 저장하며 이는 트래픽 분석 및 트러블 슈팅을 하는데 도움이 된다. Elastic Load Balancing은 5분마다 각 로드 밸런서 노드에 대한 로그 파일을 게시한다. | | |
| 진단 기준 | ▶ 양호 : 각 로드 밸런서(ALB, NLB, CLB)에 "액세스 로그(Access logs)"가 활성화되어 있는 경우   ▶ 취약 : 각 로드 밸런서(ALB, NLB, CLB)에 "액세스 로그"가 비활성화되어 있는 경우   ▶ Default Value : Disabled ※ 해당 항목은 GWLB에 해당되지 않는다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 해당 로드 밸런서(ALB, NLB, CLB)를 클릭한다.  4. 상단의 '작업(Actions)' 버튼을 클릭하여, '속성 편집(Edit attributes)'을 선택한다.  5. "액세스 로그(Access logs)" 항목의 "활성화(Enable)"를 선택한다.  6. 저장(Save) 버튼을 클릭한다.   [AWS CLI] ※ ALB, NLB의 경우  1. 다음 명령어를 실행하여 해당 로드 밸런서의 액세스 로깅을 활성화한다.  # aws elbv2 modify-load-balancer-attributes --load-balancer-arn <arn> --attributes Key=access\_logs.s3.enabled,Value=true Key=access\_logs.s3.bucket,Value=<s3 bucket name> Key=access\_logs.s3.prefix,Value=<s3 prefix>  ※ CLB의 경우 1. 다음 명령어를 실행하여 해당 로드 밸런서의 액세스 로깅을 활성화한다.  # aws elb modify-load-balancer-attributes --load-balancer-name <name> --load-balancer-attributes AccessLog={Enabled=true,S3BucketName=<s3 bucket name>,EmitInterval=integer,S3BucketPrefix=<s3 prefix>} | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 로드 밸런서(ALB, NLB, CLB)를 클릭한다.  4. 설명(Description) 탭을 선택한다.  5. 속성(Attributes)에서 "액세스 로그(Access logs)" 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] ※ ALB, NLB의 경우  1. 다음 명령어를 실행하여 각 로드 밸런서의 액세스 로깅이 활성화되어 있는지 확인한다.  # aws elbv2 describe-load-balancer-attributes --load-balancer-arn <arn>  예시는 다음과 같다.  {  "Key": "access\_logs.s3.enabled",  "Value": "true"  }  ※ CLB의 경우  1. 다음 명령어를 실행하여 각 로드 밸런서의 액세스 로깅이 활성화되어 있는지 확인한다.  # aws elb describe-load-balancer-attributes --load-balancer-name <name>  예시는 다음과 같다.  {  "LoadBalancerAttributes": {  "AccessLog": {  "Enabled": true  }  } } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/elasticloadbalancing/latest/application/load-balancer-access-logs.html https://docs.aws.amazon.com/ko\_kr/elasticloadbalancing/latest/network/load-balancer-access-logs.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/elb-logging-enabled.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-elb-5 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ELB HTTPS 리스너 설정(7.3.6) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 로드 밸런서에 HTTPS 리스너를 설정하는 것을 권장한다. | | |
| 설명 | 리스너는 구성한 프로토콜 및 포트를 통해 연결 요청을 확인하는 프로세스이다. 로드 밸런서는 리스너에 대해 정의한 규칙에 따라 요청을 라우팅하는 방법을 결정한다. 리스너는 HTTP, HTTPS 프로토콜을 지원하며, 1~65536번 포트를 지원한다. HTTPS 리스너를 생성할 경우, 로드 밸런서와 SSL 또는 TLS 세션을 시작하는 클라이언트 간에 트래픽을 암호화할 수 있다. HTTPS 리스너를 사용하기 위해서는 로드 밸런서에 한 개 이상의 SSL/TLS 서버 인증서를 반드시 배포해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 로드 밸런서(ALB, NLB, CLB)에 HTTPS 리스너가 설정되어 있는 경우   ▶ 취약 : 각 로드 밸런서(ALB, NLB, CLB)에 HTTPS 리스너가 설정되어 있지 않은 경우   ▶ Default Value : Not set ※ 해당 항목은 GWLB에 해당되지 않는다. | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 해당 로드 밸런서(ALB, NLB, CLB)를 클릭한다.  4. 리스너(Listeners) 탭을 선택한다.  5. 리스너 추가(Add listener) 버튼을 클릭한다.  6. 프로토콜(Protocol)과 포트(Port)를 "HTTPS"와 "443"으로 설정한다.  7. 기본 작업, 보안 정책, 기본 SSL 인증서를 설정한 후, 리스너 추가(Add listener) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 로드 밸런서(ALB, NLB, CLB)를 클릭한다.  4. 리스너(Listeners) 탭을 선택한다.  5. 프로토콜(Protocol)과 포트(Port)가 "HTTPS"와 "443"으로 설정된 리스너가 존재하는지 확인한다.   [AWS CLI] ※ ALB, NLB의 경우  1. 다음 명령어를 실행하여 각 로드 밸런서에 HTTPS(포트 443) 리스너가 설정되어 있는지 확인한다.  # aws elbv2 describe-listeners --load-balancer-arn <arn>  ※ CLB의 경우  1. 다음 명령어를 실행하여 각 로드 밸런서에 HTTPS(포트 443) 리스너가 설정되어 있는지 확인한다.  # aws elb describe-load-balancers --load-balancer-name <name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticloadbalancing/latest/application/create-https-listener.html https://docs.aws.amazon.com/elasticloadbalancing/latest/classic/elb-https-load-balancers.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/elb-tls-https-listeners-only.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-elb-3 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ALB 잘못된 헤더 필드 삭제 설정(7.3.7) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 유효하지 않은 헤더 필드는 라우팅하지 않고 삭제하는 것을 권장한다. | | |
| 설명 | ALB(Application Load Balancer)가 유효하지 않은 HTTP 헤더를 삭제하도록 설정하는 것을 권장한다. 잘못된 헤더 필드를 삭제할 경우, 대상으로 라우팅되지 않고 로드 밸런서에서 삭제되어 HTTP 동기화 해제 공격을 방어할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 ALB에 "잘못된 헤더 필드 삭제(Drop invalid header fields)"가 활성화되어 있는 경우   ▶ 취약 : 각 ALB에 "잘못된 헤더 필드 삭제"가 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 해당 ALB를 클릭한다.  4. 상단의 '작업(Actions)' 버튼을 클릭하여, '속성 편집(Edit attributes)'을 선택한다.  5. "잘못된 헤더 필드 삭제(Drop invalid header fields)" 항목의 "활성화(Enable)"를 선택한다.  6. 저장(Save) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 ALB의 잘못된 헤더 필드 삭제 항목을 활성화한다.  # aws elbv2 modify-load-balancer-attributes --load-balancer-arn <arn> --attributes Key=routing.http.drop\_invalid\_header\_fields.enabled,Value=true | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 ALB를 클릭한다.  4. 설명(Description) 탭을 선택한다.  5. 속성(Attributes)에서 "잘못된 헤더 필드 삭제(Drop invalid header fields)" 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 ALB의 잘못된 헤더 필드 삭제 항목이 활성화되어 있는지 확인한다.  # aws elbv2 describe-load-balancer-attributes --load-balancer-arn <arn>  예시는 다음과 같다.  {  "Key": "routing.http.drop\_invalid\_header\_fields.enabled",  "Value": "true" } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/config/latest/developerguide/alb-http-drop-invalid-header-enabled.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-elb-4 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ALB HTTP-HTTPS 리다이렉션 설정(7.3.8) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 전송 중 암호화를 위해 클라이언트 HTTP 요청을 HTTPS 요청으로 리다이렉션하는 것을 권장한다. | | |
| 설명 | ALB(Application Load Balancer)의 모든 HTTP 리스너에 HTTP-HTTPS 리다이렉션이 구성되어 있는지 확인하는 것을 권장한다. 리스너는 구성한 프로토콜 및 포트를 통해 연결 요청을 확인하는 프로세스이다. 로드 밸런서는 리스너에 대해 정의한 규칙에 따라 요청을 라우팅하는 방법을 결정한다. 리스너는 HTTP, HTTPS 프로토콜을 지원하며, 1~65536번 포트를 지원한다. HTTPS 리스너를 사용하여 ALB로 암호화 및 복호화 작업을 오프로드할 수 있으며, 로드 밸런서와 SSL 또는 TLS 세션을 시작하는 클라이언트 간에 트래픽을 암호화할 수 있다. 전송 중 암호화를 위해 클라이언트 HTTP 요청을 포트 443의 HTTPS 요청으로 리다이렉션하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 ALB의 HTTP 리스너에 HTTPS 443 포트로 리다이렉션하는 규칙이 설정되어 있는 경우    ▶ 취약 : 각 ALB의 HTTP 리스너에 HTTPS 443 포트로 리다이렉션하는 규칙이 설정되어 있지 않은 경우   ▶ Default Value : Not set | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 해당 ALB를 클릭한다.  4. 리스너(Listeners) 탭을 선택한다.  5. "HTTP:80" 리스너를 체크한 후, '편집(Edit)' 버튼을 클릭한다.  6. 기본 작업(Default action)에 존재하는 기존 규칙을 삭제한다.  7. 작업 추가(Add action)를 클릭하여, "리디렉션 대상(Redirect to...)"을 선택한다.  8. 프로토콜에 "HTTPS"을 설정하고, 포트에 "443"을 입력한다.  9. 하단의 확인 아이콘을 클릭한다.  10. 상단의 '업데이트(Update)' 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 ALB를 클릭한다.  4. 리스너(Listeners) 탭을 선택한다.  5. "HTTP:80" 리스너에 HTTPS 443 포트로 리다이렉션하는 규칙이 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 ALB의 "HTTP:80" 리스너에 HTTPS 443 포트로 리다이렉션하는 규칙이 설정되어 있는지 확인한다.  # aws elbv2 describe-listeners --load-balancer-arn<arn>  예시는 다음과 같다.  {  "Listeners": [  {  "ListenerArn": "<listener arn>",  "LoadBalancerArn": "<load balancer arn>",  "Port": 80,  "Protocol": "HTTP",  "DefaultActions": [  {  "Type": "redirect",  "Order": 1,  "RedirectConfig": {  "Protocol": "HTTPS",  "Port": "443",  "Host": "#{host}",  "Path": "/#{path}",  "Query": "#{query}",  "StatusCode": "HTTP\_301"  } } ]  }  ] } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/config/latest/developerguide/alb-http-to-https-redirection-check.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-elbv2-1 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | ALB에 WAF 활성화(7.3.9) | 위험도 | High |
| 대분류 | Networking | | |
| 중분류 | ELB | | |
| 요약 | 웹 서비스 보호를 위해 ALB에 WAF를 활성화하는 것을 권장한다. | | |
| 설명 | WAF(Web Application Firewall)는 애플리케이션의 가용성에 영향을 미치거나 과도한 자원을 소비할 수 있는 외부 공격을 보호하는 서비스로 웹 애플리케이션의 특정 경로에 대한 접근을 허용 또는 거부할 수 있으며, 다양한 SQL 인젝션 공격을 차단할 수 있다. WAF의 접근 제어 목록(ACL)을 생성한 후 ALB(Application Load Balancer)를 상호 연동하여, 차단 또는 허용할 규칙을 생성해 내부 및 외부 응용 프로그램과 웹 서비스를 모두 보호할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 ALB에 WAF가 활성화되어 있는 경우  ▶ 취약 : 각 ALB에 WAF가 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/wafv2/ 를 통해 AWS WAF에 접속한다.  2. 좌측 사이드 바에서 'Web ACLs'을 클릭한다. 3. ALB에 연결할 web ACL을 클릭한다.  4. 'Associated AWS resources' 탭을 선택한다.  5. 'Add AWS resources' 버튼을 클릭한다.  6. 'Resource type'에서 "Application Load Balancer"를 선택한다.  7. 연결할 ALB를 선택한다.  8. 'Add' 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 ALB에 WAF를 활성화한다.  # aws wafv2 associate-web-acl --web-acl-arn <web acl arn> --resource-arn <resource arn> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/ec2/ 를 통해 EC2 대시보드에 접속한다. 2. 좌측 사이드 바에서 '로드 밸런서(Load Balancers)'를 클릭한다.  3. 각 ALB를 클릭한다.  4. 통합 서비스(Integrated services) 탭을 선택한다. 5. "AWS WAF" 서비스가 활성화되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 ALB에 WAF가 활성화되어 있는지 확인한다.  # aws wafv2 get-web-acl-for-resource --resource-arn <arn> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://aws.amazon.com/ko/blogs/korea/aws-web-application-firewall-waf-for-application-load-balancers/ https://docs.aws.amazon.com/config/latest/developerguide/alb-waf-enabled.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Route53 도메인 자동 갱신 활성화(7.4.1) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | Route 53 | | |
| 요약 | Route53 도메인의 자동 갱신을 활성화하는 것을 권장한다. | | |
| 설명 | 도메인의 자동 갱신(Auto renew)이 비활성화되어 있는 경우, 기존에 서비스되는 웹 페이지의 연결된 도메인이 만료되어 사용자가 해당 서비스를 이용할 수 없다. | | |
| 진단 기준 | ▶ 양호 : 각 Route53 도메인에 "자동 갱신(Auto renew)"이 활성화되어 있는 경우   ▶ 취약 : 각 Route53 도메인에 "자동 갱신"이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/route53 을 통해 AWS Route 53에 접속한다. 2. 좌측 사이드 바에서 '등록된 도메인(Registered domains)'을 클릭한다.  3. 해당 도메인을 클릭한다.  4. '자동 갱신(Auto renew)' 항목을 "활성(Enabled)"으로 설정한다.   [AWS CLI] 1. 다음 명령어를 실행하여 자동 갱신을 활성화한다.  # aws route53domains enable-domain-auto-renew --domain-name <domain name> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/route53 을 통해 AWS Route 53에 접속한다. 2. 좌측 사이드 바에서 '등록된 도메인(Registered domains)'을 클릭한다.  3. 각 도메인을 클릭한다.  4. '자동 갱신(Auto renew)' 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 자동 갱신이 활성화되어 있는지 확인한다.  # aws route53domains list-domains --region us-east-1 ※ 리전명을 us-west-1 로 명시해주어야 한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/Route53/latest/DeveloperGuide/domain-renew.html https://docs.aws.amazon.com/ko\_kr/Route53/latest/DeveloperGuide/domain-enable-disable-auto-renewal.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Route53 도메인 이전 잠금 설정(7.4.2) | 위험도 | Medium |
| 대분류 | Networking | | |
| 중분류 | Route 53 | | |
| 요약 | 다른 등록 대행자로의 무단 이전을 방지하기 위해 도메인을 잠그는 것을 권장한다. | | |
| 설명 | 악의적인 사용자가 무단으로 도메인을 다른 등록 대행자로 이전하지 못하도록 도메인 이전 잠금(Transfer lock) 기능을 활성화하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 Route53 도메인에 "이전 잠금(Transfer lock)"이 활성화되어 있는 경우   ▶ 취약 : 각 Route53 도메인에 "이전 잠금"이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/route53 을 통해 AWS Route 53에 접속한다. 2. 좌측 사이드 바에서 '등록된 도메인(Registered domains)'을 클릭한다.  3. 해당 도메인을 클릭한다.  4. '이전 잠금(Transfer lock)' 항목을 "활성(Enabled)"으로 설정한다.   [AWS CLI] 1. 다음 명령어를 실행하여 이전 잠금을 활성화한다.  # aws route53domains enable-domain-transfer-lock --domain-name <domain name> | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/route53 을 통해 AWS Route 53에 접속한다. 2. 좌측 사이드 바에서 '등록된 도메인(Registered domains)'을 클릭한다.  3. 각 도메인을 클릭한다.  4. '이전 잠금(Transfer lock)' 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 이전 잠금이 활성화되어 있는지 확인한다.  # aws route53domains list-domains --region us-east-1 ※ 리전명을 us-west-1 로 명시해주어야 한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/Route53/latest/DeveloperGuide/domain-lock.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | API 게이트웨이 캐싱 활성화(7.5.1) | 위험도 | Low |
| 대분류 | Networking | | |
| 중분류 | API Gateway | | |
| 요약 | REST API의 성능을 최적화하고 요청에 대한 빠른 응답을 위해 캐싱을 활성화하는 것을 권장한다. | | |
| 설명 | API 게이트웨이에서 API 캐싱을 활성화하여 엔트포인트의 응답을 캐싱할 수 있다. 스테이지에 대한 캐싱을 활성화할 경우, API 게이트웨이에 지정된 TTL(Time-to-Live) 기간(초) 동안 엔드포인트에 대한 응답을 캐싱하여 캐시에서 엔드포인트 응답을 조회하여 요청에 응답한다. API 캐싱에 대한 기본 TTL 값은 300초이며 최대 TTL 값은 3600초이다. TTL이 0일 경우 캐싱이 비활성화된다.   ※ 캐싱을 활성화할 경우, 기본적으로 GET 메서드만 캐싱을 활성화한다. 다른 메서드의 캐싱을 활성화하기 위해서는 메서드 설정을 재정의해야 한다.  ※ 캐싱은 시간 단위로 요금이 청구된다.  ※ 캐시 용량은 캐시 인스턴스의 CPU, 메모리 및 네트워크 대역폭에 영향을 미칠 수 있다.  ※ 캐시 데이터를 암호화할 경우, 응답의 크기가 증가할 수 있다. | | |
| 진단 기준 | ▶ 양호 : 각 REST API에 API 캐시가 활성화되어 있고, 캐시 데이터가 암호화되어 있는 경우  ▶ 취약 : 각 REST API에 API 캐시가 활성화되어 있지 않은 경우 또는 캐시 데이터가 암호화되어 있지 않은 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/apigateway/ 를 통해 Amazon API Gateway에 접속한다.  2. 해당 REST API를 클릭한다.  3. 좌측 사이드 바에서 '스테이지(Stages)'를 클릭한다. 4. 해당 스테이지를 클릭한다.  5. 설정(Settings) 탭을 선택한다.  6. 캐시 설정(Cache Settings)에서 "API 캐시 활성화(Enable API cache)" 항목을 체크한다. 7. 캐시 용량, 캐시 TTL을 설정하고, "캐시 데이터 암호화(Encrypt cache data)" 항목을 체크한다.  8. 변경 사항 저장(Save changes) 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/apigateway/ 를 통해 Amazon API Gateway에 접속한다.  2. 각 REST API를 클릭한다.  3. 좌측 사이드 바에서 '스테이지(Stages)'를 클릭한다. 4. 각 스테이지를 클릭한다.  5. 설정(Settings) 탭을 선택한다.  6. 캐시 설정(Cache Settings)에서 "API 캐시 활성화(Enable API cache)" 항목이 활성화되어 있는지 확인한다. 7. "캐시 데이터 암호화(Encrypt cache data)" 항목이 활성화되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 API 게이트웨이의 스테이지에 API 캐시 활성화, 캐시 데이터 암호화가 활성화되어 있는지 확인한다.  # aws apigateway get-stage --rest-api-id <api gw id> --stage-name <stage name> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/apigateway/latest/developerguide/api-gateway-caching.html https://docs.aws.amazon.com/config/latest/developerguide/api-gw-cache-enabled-and-encrypted.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html | | |

**Analytics**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Elasticsearch 도메인 퍼블릭 액세스 노출 제한(8.1.1) | 위험도 | High |
| 대분류 | Analytics | | |
| 중분류 | Elasticsearch Service | | |
| 요약 | Elasticsearch 도메인이 외부에 노출되는 것을 방지해야 한다. | | |
| 설명 | 퍼블릭 액세스를 허용 및 모든 사용자 또는 0.0.0.0/0에서 접근 가능하도록 설정할 경우 어떠한 접근 제어 없이 인터넷에 노출되는 것과 같으며, 이는 데이터 유출로 연결될 수 있다. 엘라스틱서치의 액세스 정책인 IAM, IP, 리소스 기반 등을 설정하여 특정 사용자 또는 특정 IP에 한하여 접근할 수 있도록 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 네트워크 구성이 VPC 액세스로 설정 또는 특정 사용자나 IP 주소에서만 액세스하도록 설정되어 있는 경우   ▶ 취약 : 네트워크 구성이 퍼블릭 액세스(public access)로 설정 및 모든 사용자 또는 0.0.0.0/0에서 액세스하도록 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 해당 도메인을 클릭한다. 3. VPC 탭을 클릭한다.  4. 보안 그룹(Security Groups)을 클릭하여 해당 보안 그룹으로 이동한다.  5. 인바운드 규칙(Inbound rules)과 아웃바운드 규칙(Outbound rules)에서 AWS 모든 사용자 또는 0.0.0.0/0에서 접근 가능하도록 설정하는 정책을 삭제한다.  ※ 사용자 지정 액세스 정책일 경우 해당 요소를 제거한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 각 도메인을 클릭한다. 3. VPC 탭을 클릭한다.  4. 보안 그룹(Security Groups)을 클릭하여 해당 보안 그룹으로 이동한다.  5. 인바운드 규칙(Inbound rules)과 아웃바운드 규칙(Outbound rules)에서 AWS 모든 사용자 또는 0.0.0.0/0 에서 접근 가능하도록 설정하는 정책이 없는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/elasticsearch-service/latest/developerguide/es-createupdatedomains.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Elasticsearch 노드 간 암호화 설정(8.1.2) | 위험도 | Medium |
| 대분류 | Analytics | | |
| 중분류 | Elasticsearch Service | | |
| 요약 | Elasticsearch 도메인에서 노드 간 암호화를 설정하는 것을 권장한다. | | |
| 설명 | Elasticsearch 도메인은 VPC 액세스 활성화 여부에 관계없이 자체 전용 VPC 내에 존재한다. 이를 통해 잠재적 공격자가 Elasticsearch 노드 간 트래픽을 가로채는 것을 방지하고 클러스터를 안전하게 유지할 수 있다. 하지만 기본적으로 VPC 내에서는 암호화되지 않기 때문에, ES(Elasticsearch Service) 도메인에서 노드 간 암호화를 설정하는 것을 권장한다. 노드 간 암호화를 설정할 경우, VPC 내 모든 통신에 대해 TLS 1.2 버전으로 암호화할 수 있다.  ※ 노드 간 암호화는 새 도메인의 경우 Elasticsearch 6.0 버전 이상 필요하며, 기존 도메인의 경우 Elasticsearch 6.7 버전 이상이 필요하다.  ※ 세분화된 액세스 제어를 활성화할 경우, 노드 간 암호화 항목이 자동적으로 활성화된다. | | |
| 진단 기준 | ▶ 양호 : 각 Elasticsearch 도메인의 "노드 간 암호화(Node-to-node encryption)" 항목이 활성화되어 있는 경우   ▶ 취약 : 각 Elasticsearch 도메인의 "노드 간 암호화" 항목이 비활성화되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 해당 도메인을 클릭한다. 3. 상단의 '작업(Actions)' 버튼을 클릭하여 '암호화 수정(Modify access policy)'을 선택한다. 4. "노드 간 암호화(Node-to-node encryption)" 항목을 체크한다.  5. 전송(Submit) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 노드 간 암호화를 활성화한다.  # aws es update-elasticsearch-domain-config --domain-name <domain name> --node-to-node-encryption-options Enabled=true | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 각 도메인을 클릭한다. 3. 개요(Overview) 탭을 클릭한다.  4. "노드 간 암호화(Node-to-node encryption)" 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 노드 간 암호화가 활성화되어 있는지 확인한다.  # aws es describe-elasticsearch-domain --domain-name <domain name>  예시는 다음과 같다.  {  "DomainStatus": {  ...  "NodeToNodeEncryptionOptions": {  "Enabled": true  },  ... } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticsearch-service/latest/developerguide/ntn.html https://docs.aws.amazon.com/elasticsearch-service/latest/developerguide/fgac.html https://docs.aws.amazon.com/cli/latest/reference/es/update-elasticsearch-domain-config.html https://docs.aws.amazon.com/config/latest/developerguide/elasticsearch-node-to-node-encryption-check.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-es-3 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Elasticsearch 유휴 시 데이터 암호화 설정(8.1.3) | 위험도 | Medium |
| 대분류 | Analytics | | |
| 중분류 | Elasticsearch Service | | |
| 요약 | Elasticsearch 도메인 내 유휴 데이터를 암호화하는 것을 권장한다. | | |
| 설명 | 악의적인 사용자의 무단 액세스를 방지하기 위해 Amazon ES(Elasticsearch Service) 도메인 내 유휴 데이터에 대해 암호화하는 것을 권장한다. 암호화를 설정할 경우, KMS(Key Management Service)를 사용하여 암호화 키를 저장 및 관리하며, AES-256 알고리즘으로 모든 인덱스(UltraWarm 저장 장치 포함), 전환 파일, 애플리케이션 디렉터리의 모든 기타 데이터, 자동 스냅샷 등을 암호화한다.  ※ Amazon ES는 대칭 고객 마스터 키(Customer-master key)만을 지원한다.  ※ 유휴 데이터 암호화는 새 도메인의 경우 Elasticsearch 5.1 버전 이상 필요하며, 기존 도메인의 경우 Elasticsearch 6.7 버전 이상이 필요하다.  ※ 세분화된 액세스 제어를 활성화할 경우, 유휴 상태 암호화 항목이 자동적으로 활성화된다. | | |
| 진단 기준 | ▶ 양호 : 각 Elasticsearch 도메인의 "유휴 상태 암호화(Encryption at rest)" 항목이 활성화되어 있는 경우   ▶ 취약 : 각 Elasticsearch 도메인의 "유휴 상태 암호화" 항목이 비활성화되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 해당 도메인을 클릭한다. 3. 상단의 '작업(Actions)' 버튼을 클릭하여 '암호화 수정(Modify access policy)'을 선택한다. 4. "유휴 시 데이터 암호화 활성화(Enable encryption of data at rest)" 항목을 체크한다.  5. 전송(Submit) 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 유휴 시 데이터 암호화를 활성화한다.  # aws es update-elasticsearch-domain-config --domain-name <domain name> --encryption-at-rest-options Enabled=true | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 각 도메인을 클릭한다. 3. 개요(Overview) 탭을 클릭한다.  4. "유휴 상태 암호화(Encryption at rest)" 항목이 "활성(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 유휴 상태 암호화가 활성화되어 있는지 확인한다.  # aws es describe-elasticsearch-domain --domain-name <domain name>  예시는 다음과 같다.  {  "DomainStatus": {  ...  "EncryptionAtRestOptions": {  "Enabled": true,  "KmsKeyId": "<arn>"  },  ... } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticsearch-service/latest/developerguide/encryption-at-rest.html https://docs.aws.amazon.com/elasticsearch-service/latest/developerguide/fgac.html https://docs.aws.amazon.com/cli/latest/reference/es/update-elasticsearch-domain-config.html https://docs.aws.amazon.com/config/latest/developerguide/elasticsearch-encrypted-at-rest.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-es-1 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Elasticsearch 도메인의 VPC 엔드포인트 설정(8.1.4) | 위험도 | Medium |
| 대분류 | Analytics | | |
| 중분류 | Elasticsearch Service | | |
| 요약 | Elasticsearch 도메인에 VPC 엔드포인트를 설정하는 것을 권장한다. | | |
| 설명 | VPC(Virtual Private Cloud)는 AWS 클라우드에서 다른 가상 네트워크와 논리적으로 분리되어 있다. Amazon ES(Elasticsearch Service) 도메인을 VPC 내에 배치할 경우, 인터넷 게이트웨이, NAT 디바이스 또는 VPN 연결 없이 VPC 내부에서 Amazon ES와 다른 서비스 간에 보안 통신이 가능하다. 보안 그룹을 통해 VPC의 네트워크 액세스를 제어할 수 있으며, 제한적 액세스 정책을 통해 추가적인 보안 계층을 제공할 수 있다. 퍼블릭 엔드포인트의 경우, 사용자가 액세스 권한을 제어할 수 있더라도 인터넷에 연결된 디바이스라면 모두 액세스 할 수 있으므로 도메인에 VPC 엔드포인트를 설정하는 것을 권장한다.  ※ 전용 테넌시를 사용하는 VPC 내에서 도메인을 실행할 수 없다. Default로 설정된 테넌시에서 VPC를 사용해야 한다. | | |
| 진단 기준 | ▶ 양호 : 각 Elasticsearch 도메인의 엔드포인트가 "VPC"로 설정되어 있는 경우   ▶ 취약 : 각 Elasticsearch 도메인의 엔드포인트가 "인터넷"으로 설정되어 있는 경우   ▶ Default Value : - | | |
| 조치 방안 | ※ Elasticsearch 도메인을 생성할 때 설정한 엔드포인트는 다시 변경할 수 없다. 새 도메인을 생성하여 수동으로 인덱싱하거나 데이터를 마이그레이션 해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/es/home 를 통해 Amazon Elasticsearch Service에 접속한다. 2. 각 도메인의 "엔드포인트(Endpoint)" 항목이 "VPC"로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 도메인의 엔드포인트가 VPC로 설정되어 있는지 확인한다.  # aws es describe-elasticsearch-domain --domain-name <domain name>  예시는 다음과 같다.  {  "DomainStatus": {  ...  "Endpoints": {  "vpc": "<vpc>"  },  ...  "VPCOptions": {  "VPCId": "<vpc id>",  "SubnetIds": [  "<subnet id>"  ],  "AvailabilityZones": [  "<az>"  ],  "SecurityGroupIds": [  "<sg>"  ]  },  ... } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/elasticsearch-service/latest/developerguide/es-vpc.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/elasticsearch-in-vpc-only.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EMR 퍼블릭 액세스 차단(8.2.1) | 위험도 | High |
| 대분류 | Analytics | | |
| 중분류 | EMR | | |
| 요약 | EMR 클러스터에 퍼블릭 액세스를 차단하는 것을 권장한다. | | |
| 설명 | Amazon EMR(Elastic MapReduce) 클러스터에 퍼블릭 액세스 차단이 설정되어 있는지 확인하는 것을 권장한다. 퍼블릭 액세스 차단을 설정할 경우, 클러스터의 보안 그룹에 모든 공용 주소(IPv4 0.0.0.0/0 또는 IPv6 ::/0)의 인바운드 트래픽을 허용하는 규칙이 있을 때 EMR 클러스터가 시작하는 것을 방지할 수 있다. 기본적으로 SSH 22번 포트를 제외한 모든 포트가 차단되며, 예외 처리할 포트 또는 포트 범위를 설정할 수 있다.  ※ 퍼블릭 액세스 차단 기능은 현재 리전의 모든 클러스터에 적용된다.  ※ 퍼블릭 액세스 차단 기능은 프라이빗 서브넷에는 적용되지 않는다. | | |
| 진단 기준 | ▶ 양호 : EMR 클러스터에 "퍼블릭 액세스 차단(Block public access)"이 설정되어 있는 경우   ▶ 취약 : EMR 클러스터에 "퍼블릭 액세스 차단"이 설정되어 있지 않은 경우   ▶ Default Value : Enabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/elasticmapreduce/ 를 통해 Amazon EMR에 접속한다.  2. 좌측 사이드 바에서 '퍼블릭 액세스 차단(Block public access)'을 클릭한다.  3. "퍼블릭 액세스 차단" 항목에서 "변경(Change)" 버튼을 클릭한다.  4. "설정(On)"을 선택한다.  5. 초록색의 체크(V) 아이콘을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EMR 클러스터에 퍼블릭 액세스 차단을 설정한다.  # aws emr put-block-public-access-configuration --block-public-access-configuration BlockPublicSecurityGroupRules=true | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/elasticmapreduce/ 를 통해 Amazon EMR에 접속한다.  2. 좌측 사이드 바에서 '퍼블릭 액세스 차단(Block public access)'을 클릭한다.  3. "퍼블릭 액세스 차단" 항목이 "설정(On)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 EMR 클러스터에 퍼블릭 액세스 차단이 설정되어 있는지 확인한다. "BlockPublicSecurityGroupRules" 항목이 "true"로 설정되어 있어야 한다.  # aws emr get-block-public-access-configuration  예시는 다음과 같다.  {  "BlockPublicAccessConfiguration": {  "BlockPublicSecurityGroupRules": true,  "PermittedPublicSecurityGroupRuleRanges": []  },  ... } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-block-public-access.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/emr-master-no-public-ip.html https://docs.aws.amazon.com/ko\_kr/securityhub/latest/userguide/securityhub-standards-fsbp-controls.html#fsbp-emr-1 | | |

**EKS**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | EKS 감사 로깅 활성화(9.1.1) | 위험도 | Medium |
| 대분류 | EKS | | |
| 중분류 | Control Plance Configuration | | |
| 소분류 | Logging | | |
| 요약 | Amazon EKS 쿠버네티스의 감사 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | 감사 로그는 Amazon EKS에서 관리하는 EKS 관리 제어 플레인 로그의 일부이다. Amazon EKS는 AWS CloudTrail과 통합되어 있으며, CloudTrail에서 EKS에 대한 모든 API 호출을 캡쳐할 수 있다. 감사 로그는 마스터 로그에 생성되어 디스크 공간을 차지할 수 있기 때문에, 너무 많은 양의 로그 정보를 기록할 경우 클러스터 노드의 가용성에 영향을 줄 수 있으므로 주의해야 한다. S3 수명 주기 기능을 통해 시간 경과에 따른 로그 축적 및 관리를 하도록 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 제어 플레인 로깅(Control Plane Logging) 내 모든 항목이 활성화되어 있는 경우   ▶ 취약 : 제어플레인 로깅 내 항목이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/eks 를 통해 Amazon EKS(Elastic Kubernetes Service)에 접속한다.  2. 좌측 사이드 바에서 '클러스터(Clusters)'를 클릭한다.  3. 해당 클러스터를 클릭한다.  4. 구성(Configuration) 탭을 선택 후, 로깅(Logging) 탭을 클릭한다.  5. 제어 플레인 로깅(Control Plane Logging) 우측의 '로깅 관리(Manage logging)' 버튼을 클릭한다.  6. 모든 항목을 "활성화됨(Enabled)"으로 설정한 후, 변경 사항 저장(Save changes)을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/eks 를 통해 Amazon EKS(Elastic Kubernetes Service)에 접속한다.  2. 좌측 사이드 바에서 Amazon EKS의 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 구성(Configuration) 탭을 선택 후, 로깅(Logging) 탭을 클릭한다.  5. 제어 플레인 로깅(Control Plane Logging)의 모든 항목(API 서버, 감사, Authenticator, 컨트롤러 관리자, 스케줄러)이 "활성화(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 "api", audit", "authenticator", "controllerManager", "scheduler" 가 모두 활성화되어 있는지 확인한다.  # aws --region "ap-northeast-2" eks describe-cluster --name "seeunkr" --query 'cluster.logging.clusterLogging[?enabled==`true`].types' | | |
| Reference | 2.1.1 Enable audit Logs (Manual) | | |
| Site | https://kubernetes.io/docs/tasks/debug-application-cluster/audit/ https://docs.aws.amazon.com/eks/latest/userguide/control-plane-logs.html https://docs.aws.amazon.com/eks/latest/userguide/logging-using-cloudtrail.html https://docs.aws.amazon.com/AmazonS3/latest/dev/object-lifecycle-mgmt.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | CMK를 사용한 Kubernetes Secrets 암호화 설정(9.2.1) | 위험도 | Medium |
| 대분류 | EKS | | |
| 중분류 | Control Plance Configuration | | |
| 소분류 | AWS Key Management(KMS) | | |
| 요약 | Amazon EKS 클러스터를 생성할 때 secret 암호 기능을 사용하여 etcd에 저장된 kubernetes의 secret을 암호화하는 것을 권장한다. | | |
| 설명 | Kubernetes는 마운트 된 볼륨을 통해 pod가 접근할 수 있는 secret을 저장할 수 있다. 현재 Kubernetes의 secret은 base64로 인코딩되어 저장되지만, 암호화가 권장한다. Amazon EKS 클러스터 버전 1.13 이상에서는 AWS KMS의 CMK(Customer Managed Keys)를 사용하여 Kubernetes 비밀 정보를 암호화(Secrets Encryption Feature)하는 기능을 지원한다. AWS KMS 키를 사용하여 Amazon EKS에 저장된 Kubernetes secret에 대한 envelope 암호화를 적용할 수 있다. 애플리케이션 계층 암호화는 "etcd"에 모두 저장된 서비스 계정 키와 같이 클러스터 작동에 필요한 암호와 사용자 정의 암호 등과 같은 중요한 데이터에 대한 추가 보안 계층을 제공한다. 이 기능을 사용하면 AWS KMS에서 관리하는 키를 사용하여 응용 프로그램 계층의 데이터를 암호화할 수 있으므로 "etcd"에 접근하기 위해 관리 권한을 획득하려고 하는 공격자로부터 보호할 수 있다. | | |
| 진단 기준 | ▶ 양호 : '암호 암호화(Secrets encryption)' 항목이 활성화되어 있는 경우  ▶ 취약 : '암호 암호화' 항목이 비활성화되어 있는 경우  ▶ Default Value : Disabled | | |
| 조치 방안 | Amazon EKS 클러스터 생성 시 '암호 암호화(Secrets encryption)' 섹션의 "KMS를 사용하여 Kubernetes 암호의 엔벨로프 암호화 활성화(Enable envelope encryption of Kubernetes secrets using KMS)" 옵션을 활성화한다. | | |
| 감사 방안 | [AWS Console] 1. https://us-west-2.console.aws.amazon.com/eks/ 를 통해 Amazon EKS에 접속한다.  2. 좌측 사이드 바에서 Amazon EKS의 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 구성(Configuration) 탭을 선택한다.  5. '암호 암호화(Secrets encryption)' 항목이 "활성화됨(Enabled)"으로 설정되어 있는지 확인한다.   [AWS CLI] 1. "encrpytionConfig" 파라미터 값이 "Enabled" 로 설정되어 있는지 확인한다. # aws eks describe-cluster --name="<cluster-name>" | | |
| Reference | 5.3.1 Ensure Kubernetes Secrets are encrypted using Customer Master Keys (CMKs) managed in AWS KMS (Automated) | | |
| Site | https://docs.aws.amazon.com/eks/latest/userguide/create-cluster.html https://eksworkshop.com/beginner/191\_secrets/ | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Control Plane Endpoint 접근 제어(9.2.2) | 위험도 | High |
| 대분류 | EKS | | |
| 중분류 | Control Plance Configuration | | |
| 소분류 | Cluster Networking | | |
| 요약 | 클러스터의 제어 플레인 액세스 권한을 인증된 IP 목록으로만 제한하기 위해 Endpoint Private Access를 사용하는 것을 권장한다. | | |
| 설명 | 인증된 네트워크는 클러스터의 제어 플레인(Control plane)에 액세스할 수 있는 제한된 범위의 IP 주소를 지정하는 방법이다. Kubernetes Engine은 TLS와 인증을 모두 사용하여 퍼블릭 인터넷에서 클러스터 제어 플레인에 안전하게 액세스 할 수 있도록 한다. 인증된 네트워크를 통해 액세스를 제한할 경우, 다음과 같은 보안 이점을 얻을 수 있다.  - 외부 공격으로부터 보호 : 인증된 네트워크는 사용자가 지정하는 특정 주소 집합에 대한 외부 액세스를 제한하여 추가적인 보안 계층을 제공한다. 이를 통해 클러스터의 인증 또는 권한 부여 메커니즘에 취약성이 있는 경우 클러스터에 대한 액세스를 보호할 수 있다.  - 내부 공격으로부터 보호 : 인증된 네트워크는 회사 내에서 마스터 인증서가 실수로 유출되는 것을 방지할 수 있다. Amazon EC2 외부 및 승인된 IP 범위 외부에서 사용된 유출된 인증서의 액세스를 거부한다. | | |
| 진단 기준 | ▶ 양호 : "endpointPublicAccess" 파라미터 값이 false, "endpointPrivateAccess" 파라미터 값이 true, "publicAccessCidrs" 파라미터 값이 CIDR IP주소 범위로 지정되어 있는 경우  ▶ 취약 : "endpointPublicAccess" 파라미터 값이 true, "endpointPrivateAccess" 파라미터 값이 false, "publicAccessCidrs" 파라미터 값이 0.0.0.0/0 으로 지정되어 있는 경우  ▶ Default Value : endpointPublicAccess = true, endpointPrivateAccess = false, publicAccessCidrs = 0.0.0.0/0 | | |
| 조치 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 "endpointPublicAccess" 파라미터 값을 "false", "endpointPrivateAccess" 파라미터 값을 "true", "publicAccessCidrs" 파라미터 값을 CIDR IP주소 범위로 설정한다.  # aws eks update-cluster-config --region <region> --name <cluster-name> --resources-vpc-config endpointPublicAccess=true, endpointPrivateAccess=true, publicAccessCidrs="<ip addres set(CIDR)>" | | |
| 감사 방안 | [AWS CLI] 1. 다음 명령어를 실행하여 "endpointPublicAccess" 파라미터 값이 "false", "endpointPrivateAccess" 파라미터 값이 "true", "publicAccessCidrs" 파라미터 값이 CIDR IP주소 범위로 지정되어 있는지 확인한다. # aws eks describe-cluster --region <region> --name <cluster-name>  결과값의 예시는 다음과 같다.  ...   "endpointPublicAccess": false,   "endpointPrivateAccess": true,   "publicAccessCidrs": [ "203.0.113.5/32" ]  ... | | |
| Reference | 5.4.1 Restrict Access to the Control Plane Endpoint (Manual) | | |
| Site | https://docs.aws.amazon.com/eks/latest/userguide/cluster-endpoint.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 클러스터 전용 보안 그룹 설정(9.2.3) | 위험도 | Low |
| 대분류 | EKS | | |
| 중분류 | Control Plance Configuration | | |
| 소분류 | Cluster Networking | | |
| 요약 | 각 클러스터마다 전용 보안 그룹을 설정하는 것을 권장한다. | | |
| 설명 | 각 클러스터마다 보안 그룹을 설정하여 다른 리소스 설정에 피해가 가지 않도록 해야 한다. 다른 EKS 클러스터 또는 리소스와 보안 그룹을 동일하게 적용할 경우 일부 리소스에 대한 연결을 차단할 가능성이 있으므로 EKS 클러스터별로 보안 그룹을 독립적으로 설정하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 클러스터마다 고유 클러스터 보안 그룹(Security Groups)으로 설정되어 있는 경우   ▶ 취약 : 클러스터마다 고유 클러스터 보안 그룹으로 설정되어 있지 않은 경우   ▶ Default Value : - | | |
| 조치 방안 | [AWS Console] 1. https://us-west-2.console.aws.amazon.com/eks/ 를 통해 Amazon EKS에 접속한다.  2. 좌측 사이드 바에서 Amazon EKS의 '클러스터(Clusters)'를 클릭한다.  3. 해당 클러스터를 클릭한다.  4. 각 클러스터마다 각기 다른 보안 그룹(Security Groups)으로 설정하도록 변경한다. | | |
| 감사 방안 | [AWS Console] 1. https://us-west-2.console.aws.amazon.com/eks/ 를 통해 Amazon EKS에 접속한다.  2. 좌측 사이드 바에서 Amazon EKS의 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 구성(Configuration) 탭을 클릭하여 네트워킹(Networking) 탭을 선택한다. 5. 클러스터 보안 그룹(Security Groups)이 해당 클러스터에서만 사용 중인지 확인한다.  [AWS CLI] 1. 다음 명령어를 실행하여 클러스터의 보안 그룹을 확인한다. # aws eks describe-cluster --name <cluster\_name> --query cluster.resourcesVpcConfig.securityGroupIds | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/eks/latest/userguide/sec-group-reqs.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | 클러스터 보안 그룹 규칙 중 인터넷 노출 제한(9.2.4) | 위험도 | High |
| 대분류 | EKS | | |
| 중분류 | Control Plance Configuration | | |
| 소분류 | Cluster Networking | | |
| 요약 | 클러스터 보안 그룹 규칙 중 인터넷에 노출된 규칙이 존재하지 않아야 한다. | | |
| 설명 | 클러스터 보안 그룹 중 인터넷에 노출된 규칙이 있는 경우, AWS IAM과 native Kubernetes RBAC로 접근 제어가 된 API 서버 엔드포인트가 인터넷에 공개된 것을 의미한다. 이는 클러스터가 외부에 공개되어 기존 인프라에 접근할 필요없이 악의적인 사용자가 직접적으로 접근할 수 있다는 것을 의미한다. | | |
| 진단 기준 | ▶ 양호 : API 서버 엔드포인트 엑세스(API server endpoint access) 가 "Private"으로 설정되어 있거나 클러스터 보안 그룹 규칙(Cluster security group) 중 소스가 0.0.0.0/0 이고 특정 포트로 설정되어 있지 않은 경우   ▶ 취약 : API 서버 엔드포인트 액세스가 "Public"으로 설정되어 있거나 클러스터 보안 그룹 규칙 중 소스가 0.0.0.0/0 이고 특정 포트로 설정되어 있는 경우   ▶ Default Value : API server endpoint access = public | | |
| 조치 방안 | [AWS Console] 1. https://us-west-2.console.aws.amazon.com/eks/ 를 통해 Amazon EKS에 접속한다.  2. 좌측 사이드 바에서 Amazon EKS의 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 구성(Configuration) 탭을 클릭하여 네트워킹(Networking) 탭을 선택한다. 5. 클러스터 보안 그룹(Cluster security group)을 클릭하여, 보안 그룹 ID(Security group ID)를 선택한다.  6. 인바운드 규칙 편집(Edit inbound rules) 버튼을 클릭한다. 7 해당 규칙의 '삭제(delete)' 버튼을 클릭한 후, '규칙 저장(Save rules)' 버튼을 클릭한다. | | |
| 감사 방안 | [AWS Console] 1. https://us-west-2.console.aws.amazon.com/eks/ 를 통해 Amazon EKS에 접속한다.  2. 좌측 사이드 바에서 Amazon EKS의 '클러스터(Clusters)'를 클릭한다.  3. 각 클러스터를 클릭한다.  4. 구성(Configuration) 탭을 클릭하여 네트워킹(Networking) 탭을 선택한다. 5. 클러스터 보안 그룹(Cluster security group) 을 클릭하여, 보안 그룹 ID(Security group ID)를 선택한다.  6. 인바운드 규칙(Inbound rules) 중 0.0.0.0/0 에서 특정 포트로 설정된 규칙이 없는지 확인한다. | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/ko\_kr/eks/latest/userguide/sec-group-reqs.html | | |

**Security&Compliance**

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | Security Hub 활성화(10.2.1) | 위험도 | Low |
| 대분류 | Security&Compliance | | |
| 중분류 | Security Hub | | |
| 요약 | AWS 리소스의 보안 상태를 주기적으로 점검하기 위해 모든 리전에 Security Hub를 활성화하는 것을 권장한다. | | |
| 설명 | Security Hub는 자동 보안 검사를 실행하여 AWS 리소스의 보안 상태에 대한 포괄적인 견해를 제공하고, 보안 업계 규정 및 모범 사례를 바탕으로 리소스 환경을 점검할 수 있도록 도와준다. 또한, AWS 계정, 서비스 및 지원되는 타사 파트너 제품 전체에서 보안 데이터를 수집하여 보안 추세를 분석하고 우선 순위가 가장 높은 보안 문제를 파악할 수 있다. 주기적인 리소스 보안 상태 점검을 위하여 Security Hub를 활성화하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : Security Hub가 모든 리전에 활성화되어 있는 경우   ▶ 취약 : Security Hub가 모든 리전에 활성화되어 있지 않은 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/securityhub/ 를 통해 AWS Security Hub에 접속한다.  2. "Security Hub로 이동(Go to Security Hub)" 버튼을 클릭한다.  3. Security Hub를 통해 점검할 보안 표준을 알맞게 선택한 후, 하단의 'Security Hub 활성화(Enable Security Hub)' 버튼을 클릭한다.  ※ 모든 리전에 Security Hub를 활성화한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Security Hub를 활성화한다.  # aws securityhub enable-security-hub ※ 모든 리전에 Security Hub를 활성화한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/securityhub/ 를 통해 AWS Security Hub에 접속한다.  2. Security Hub가 활성화되어 있는지 확인한다. ※ 모든 리전에 Security Hub가 활성화되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Security Hub가 활성화되어 있는지 확인한다.  # aws securityhub describe-hub ※ 모든 리전에 Security Hub가 활성화되어 있는지 확인한다.   예시는 다음과 같다.  {  "HubArn": "<arn>",  "SubscribedAt": "2021-07-22T05:14:34.378Z",  "AutoEnableControls": boolean } | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/securityhub/latest/userguide/what-is-securityhub.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/securityhub-enabled.html | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 타이틀(Key) | WAF의 Web ACL 로깅 활성화(10.3.1) | 위험도 | Medium |
| 대분류 | Security&Compliance | | |
| 중분류 | WAF | | |
| 요약 | WAF의 Web ACL에 대한 로깅을 활성화하는 것을 권장한다. | | |
| 설명 | AWS WAF(Web Application Firewall)는 Amazon CloudFront, Amazon API Gateway REST API, 애플리케이션 로드 밸런서 또는 AWS AppSync GraphQL API로 전달되는 HTTP 및 HTTPS 요청을 모니터링 할 수 있는 웹 애플리케이션 방화벽이다. WAF는 Web ACL을 통해 보호된 리소스가 응답하는 웹 요청에 대해 세부적인 제어가 가능하며, 조건(요청 IP 주소, 요청 국가, 요청에 포함된 문자열, 악성 SQL 코드 등)을 설정하여 요청을 허용 또는 차단할 수 있다. Web ACL에 대한 로깅을 활성화하여 요청한 사용자가 루트인지 또는 IAM 사용자인지, 임시 보안 자격 증명을 사용하여 생성된 요청인지 파악하는 것을 권장한다. | | |
| 진단 기준 | ▶ 양호 : 각 Web ACL의 로깅이 활성화되어 있는 경우   ▶ 취약 : 각 Web ACL의 로깅이 비활성화되어 있는 경우   ▶ Default Value : Disabled | | |
| 조치 방안 | [AWS Console] 1. https://console.aws.amazon.com/wafv2/ 를 통해 AWS WAF에 접속한다.  2. 좌측 사이드 바에서 'Web ACLs'를 클릭한다.  3. 리전을 설정한 후, 각 Web ACL을 클릭한다.  4. 'Logging and metrics' 탭을 선택한다.  5. "Enable logging" 버튼을 클릭한다.  6. Delivery stream, Redacted fields, Filter logs 를 알맞게 설정한 후, Save 버튼을 클릭한다.   [AWS CLI] 1. 다음 명령어를 실행하여 Web ACL의 로깅을 활성화한다.  # aws wafv2 put-logging-configuration --logging-configuration ResourceArn=<web acl arn>,LogDestinationConfigs=<kinesis firehose arn> --region <region> ※ Kinesis Data Firehose delivery streams의 이름은 "aws-waf-logs-"로 시작해야 한다. | | |
| 감사 방안 | [AWS Console] 1. https://console.aws.amazon.com/wafv2/ 를 통해 AWS WAF에 접속한다.  2. 좌측 사이드 바에서 'Web ACLs'를 클릭한다.  3. 리전을 설정한 후, 각 Web ACL을 클릭한다.  4. 'Logging and metrics' 탭을 선택한다.  5. "Logging" 항목이 "Enabled"로 설정되어 있는지 확인한다.   [AWS CLI] 1. 다음 명령어를 실행하여 각 Web ACL의 로깅이 활성화되어 있는지 확인한다.  # aws wafv2 get-logging-configuration --resource-arn <wel acl arn> --region <region> | | |
| Reference | Tatum\_bestpractice | | |
| Site | https://docs.aws.amazon.com/waf/latest/APIReference/Welcome.html https://docs.aws.amazon.com/waf/latest/developerguide/waf-chapter.html https://docs.aws.amazon.com/config/latest/developerguide/evaluate-config\_use-managed-rules.html https://docs.aws.amazon.com/config/latest/developerguide/wafv2-logging-enabled.html https://docs.aws.amazon.com/cli/latest/reference/wafv2/get-web-acl.html | | |