AWS Secure Environment Accelerator

Config File Sample Snippets (Parameters not in sample config files)

- Tweak Interface Endpoint security groups

SEA v1.3.3 locked down interface endpoint security groups to 0.0.0.0/0:443 inbound, no outbound-rules

- Some endpoints may require additional inbound ports
- these can be specified by adding the following to the config file, for each specific interface endpoint
- this setting overides the default port 443 for the specified endpoint(s) only
- the below example overides the sg for the logs endpoint and the ssmmessages endpoints on all vpcs with endpoints

In global-options:

```
"endpoint-port-orverides": {
  "logs": ["TCP:443", "UDP:9418"],
  "ssmmessages": ["TCP:443", "TCP:8080"]
}
```

- additionally customers can lock down the endpoints on each vpc to specific CIDR ranges

In vpc section, under interface endpoints:

```
"interface-endpoints": {
        "allowed-cidrs": ["10.0.0.0/8", "100.96.252.0/23", "100.96.250.0/23"]
```

- Create a role with trust policies

```
{
            "role": "Demo-Role",
            "type": "other",
            "policies": ["AdministratorAccess"],
            "boundary-policy": "Default-Boundary-Policy",
            "source-account": "operations",
            "source-account-role": "TempAdmin",
            "trust-policy": "none"
}
```

- Manage account level SCPs

- Until v1.3.3, SEA only managed SCPs on the top level OUs, where the ability to manage account level SCPs was added
- If no account level SCP settings exist, account SCPs remain managed through AWS orgs
- If an account level SCP setting exists, it enforces the SCPs on the account to be as specified in the config file

```
"fun-acct": {
  "account-name": "TheFunAccount",
  "email": "myemail+aseaT-funacct@example.com",
  "src-filename": "config.json",
  "scps": ["Guardrails-Part-2", "Guardrails-Sandbox"],
  "ou": "Sandbox"
}
```

- Creates DNS query logging and associate to the VPC

```
"vpc": {
  "dns-resolver-logging": true
}
```

- Update Central Logging Kinesis stream shard count as accounts are added

```
"central-log-services": {
  "kinesis-stream-shard-count": 2
}
```

- CWL retention values

`"default-cwl-retention"` valid values are one of: [1, 3, 5, 7, 14, 30, 60, 90, 120, 150, 180, 365, 400, 545, 731, 1827, 3653]

- Override default CWL retention period (Add to any account)

```
 "cwl-retention": 180
```

- Valid options for vpc flow logs setting on each VPC

```
"flow-logs": "S3"  ---> S3, CWL, BOTH, NONE
```

- Macie Frequency Supported Values:

```
"macie-frequency": "SIX_HOURS" ---> FIFTEEN_MINUTES, ONE_HOUR, SIX_HOURS
```

- Control MAD/ADC AZ’s:

- if not specified and more than 2 az’s exist, selects the first two defined az’s in the subnet

```
"azs": ["a", "d"]
```

- CWL subscription exclusions example

```
"central-log-services": {
  "cwl-glbl-exclusions": ["/xxx/yyy/*", "abc/*"],
  "cwl-exclusions": [
    {
      "account": "fun-acct",
      "exclusions": ["def/*"]
    }
  ]
}
```

- Add a policy to a role in the account to enable RO access to the Log Archive bucket

```
"ssm-log-archive-read-only-access": true
```

- CloudWatch Metric Filters and Alarms

```
"global-options": {
    "cloudwatch": {
      "metrics": [{
          "filter-name": "SecurityGroupChangeMetricTest",
          "accounts": [
            "management"
          ],
          "regions": [
            "${HOME_REGION}"
          ],
          "loggroup-name": "/PBMMAccel/CloudTrail",
          "filter-pattern": "{ ($.eventName = AuthorizeSecurityGroupIngress) || ($.eventName = Authorize
          "metric-namespace": "CloudTrailMetrics",
          "metric-name": "SecurityGroupEventCountTest",
          "metric-value": "1",
          "default-value": 0
      }],
      "alarms": {
        "default-accounts": [
          "management"
        ],
        "default-regions": [
          "${HOME_REGION}"
        ],
        "default-namespace": "CloudTrailMetrics",
        "default-statistic": "Sum",
        "default-period": 300,
        "default-threshold-type": "Static",
        "default-comparison-operator": "GreaterThanOrEqualToThreshold",
        "default-threshold": 1,
        "default-evaluation-periods": 1,
        "default-treat-missing-data": "notBreaching",
        "definitions": [{
            "alarm-name": "AWS-Security-Group-Changed",
            "metric-name": "SecurityGroupEventCount",
            "sns-alert-level": "Low",
            "alarm-description": "Alarms when one or more API calls are made to create, update or delete
        }]
      }
    }
  }
```

- Additional regions for Amazon CloudWatch Central Logging to S3

```
"additional-cwl-regions": {
  "${GBL_REGION}": {
    "kinesis-stream-shard-count": 1
  }
}
```

- SNS Topics - If section not provided we create High,Medium,Low,Ignore SNS topics without subscribers

```
"central-log-services": {
  "sns-excl-regions": ["sa-east-1"],
  "sns-subscription-emails": {
    "High": ["notify+high@example.com"],
    "Low": ["notify+low@example.com"],
    "Medium": ["notify+medium@example.com"]
  }
}
```

- Cert REQUEST format (Import shown in sample)

```
  "certificates": [
    {
      "name": "PublicCert",
      "type": "request",
      "domain": "*.example.com",
      "validation": "DNS",
      "san": ["*.example1.com"]
    }
  ]
```

- Other Budget ”include” fields

```
"default-budgets": {
  "name": "Default Core Budget",
  "period": "Monthly",
  "amount": 1000,
  "include": [
    "Refunds",
    "Credits",
    "Upfront-reservation-fees",
    "Recurring-reservation-charges",
    "Other-subscription-costs",
    "Taxes",
    "Support-charges",
    "Discounts"
  ]
}
```

- Cross Account Role Example

```
{
  "role": "Test-Role",
  "type": "account",
  "policies": ["AdministratorAccess"],
  "boundary-policy": "Default-Boundary-Policy",
  "source-account": "security",
  "source-account-role": "AWSLandingZoneSecurityAdministratorRole",
  "trust-policy": "role-trust-policy.txt"
}
```

- Very basic workload account example and ”per account” exceptions example

```
"workload-account-configs": {
  "fun-acct": {
    "account-name": "TheFunAccount",
    "email": "myemail+pbmmT-funacct@example.com---------------------REPLACE----------------------",
    "ou": "Sandbox",
    "exclude-ou-albs": true
  },
  "mydevacct1": {
    "account-name": "MyDev1",
    "email": "myemail+pbmmT-dev1@example.com---------------------REPLACE----------------------",
    "ou": "Dev",
    "share-mad-from": "operations",
    "enable-s3-public-access": true,
    "keep-default-vpc-regions": []
  }
}
```

- Sample limit increases supported

```
"limits": {
  "Amazon VPC/Interface VPC endpoints per VPC": 90,
  "Amazon VPC/VPCs per Region": 15,
  "AWS CloudFormation/Stack count": 400,
  "AWS CloudFormation/Stack sets per administrator account": 400
}
```

- v1.0.4_to_v1.0.5 upgrade MAD fix - REQUIRED ALL 1.0.4 ORIGINAL INSTALLS

```
"deployments": {
  "mad": {
    "password-secret-name": "accelerator/operations/mad/password"
  }
}
```

- Sample Complex Security Group

```
    {
      "name": "SampleComplexSecurityGroup",
      "inbound-rules": [
        {
          "description": "Allow Inbound Domain Traffic",
          "tcp-ports": [464, 389, 3389, 445, 88, 135, 636, 53],
          "udp-ports": [445, 138, 464, 53, 389, 123, 88],
          "source": ["0.0.0.0/0"]
        },
        {
          "description": "Allow Inbound RDSH",
          "type": ["TCP"],
          "toPort": 3269,
          "fromPort": 3268,
          "source": ["0.0.0.0/0"]
        },
        {
          "description": "Allow Inbound High Ports",
          "type": ["TCP"],
          "toPort": 65535,
          "fromPort": 1024,
          "source": ["0.0.0.0/0"]
        }
      ],
      "outbound-rules": [
        {
          "description": "All Outbound",
          "type": ["ALL"],
          "source": ["0.0.0.0/0", "0::/0"]
```

```
              }
            ]
          },
```

- Sample Single AZ NATGW

```
"natgw": {
    "subnet": {
      "name": "Web",
      "az": "a"
    }
  },
  "subnets": [
    {
      "name": "Web",
      "share-to-ou-accounts": false,
      "share-to-specific-accounts": [],
      "definitions": [
        {
          "az": "a",
          "route-table": "SandboxVPC_IGW",
          "cidr": "10.6.32.0/20"
        },
        {
          "az": "b",
          "route-table": "SandboxVPC_IGW",
          "cidr": "10.6.128.0/20"
        }
      ]
    },
    {
      "name": "App",
      "share-to-ou-accounts": false,
      "share-to-specific-accounts": [],
      "definitions": [
        {
          "az": "a",
          "route-table": "SandboxVPC_Common",
          "cidr": "10.6.0.0/19"
        },
        {
          "az": "b",
          "route-table": "SandboxVPC_Common",
          "cidr": "10.6.96.0/19"
        }
      ]
    },
    {
      "name": "Data",
      "share-to-ou-accounts": false,
      "share-to-specific-accounts": [],
      "definitions": [
        {
          "az": "a",
          "route-table": "SandboxVPC_Common",
          "cidr": "10.6.48.0/20"
        },
        {
          "az": "b",
          "route-table": "SandboxVPC_Common",
          "cidr": "10.6.144.0/20"
        }
      ]
    }
  ],
  "route-tables": [
    {
      "name": "SandboxVPC_IGW",
      "routes": [
        {
          "destination": "0.0.0.0/0",
          "target": "IGW"
        }
      ]
    },
    {
      "name": "SandboxVPC_Common",
      "routes": [
        {
          "destination": "0.0.0.0/0",
          "target": "NATGW_Web_azA"
        }
      ]
    }
  ]
```

- Sample PER AZ NATGW

```
"natgw": {
    "subnet": {
      "name": "Web"
    }
  },
  "subnets": [
    {
      "name": "Web",
      "share-to-ou-accounts": false,
      "share-to-specific-accounts": [],
      "definitions": [
        {
          "az": "a",
          "route-table": "SandboxVPC_IGW",
          "cidr": "10.6.32.0/20"
        },
        {
          "az": "b",
          "route-table": "SandboxVPC_IGW",
          "cidr": "10.6.128.0/20"
        }
      ]
    },
    {
      "name": "App",
      "share-to-ou-accounts": false,
      "share-to-specific-accounts": [],
      "definitions": [
        {
          "az": "a",
          "route-table": "SandboxVPC_a",
          "cidr": "10.6.0.0/19"
        },
        {
          "az": "b",
          "route-table": "SandboxVPC_b",
          "cidr": "10.6.96.0/19"
        }
      ]
    },
    {
      "name": "Data",
      "share-to-ou-accounts": false,
      "share-to-specific-accounts": [],
      "definitions": [
        {
          "az": "a",
          "route-table": "SandboxVPC_a",
          "cidr": "10.6.48.0/20"
        },
        {
          "az": "b",
          "route-table": "SandboxVPC_b",
          "cidr": "10.6.144.0/20"
        }
      ]
    }
  ],
  "route-tables": [
    {
      "name": "SandboxVPC_IGW",
      "routes": [
        {
          "destination": "0.0.0.0/0",
          "target": "IGW"
        }
      ]
    },
    {
      "name": "SandboxVPC_a",
      "routes": [
        {
          "destination": "0.0.0.0/0",
          "target": "NATGW_Web_azA"
        }
      ]
```

```
  },
  {
    "name": "SandboxVPC_b",
    "routes": [
      {
        "destination": "0.0.0.0/0",
        "target": "NATGW_Web_azB"
      }
    ]
  }
]
```

- TGW Route tables plus Multiple TGWs

```
"tgw": [
  {
    "name": "Main",
    "asn": 65521,
    "region": "${HOME_REGION}",
    "features": {
      "DNS-support": true,
      "VPN-ECMP-support": true,
      "Default-route-table-association": false,
      "Default-route-table-propagation": false,
      "Auto-accept-sharing-attachments": true
    },
    "route-tables": ["core", "segregated", "shared", "standalone"],
    "tgw-routes": [
      {
        "name": "{TGW_ALL}",
        "routes": [
          {
            "destination": "1.1.0.0/32",
            "target-tgw": "East"
          }
        ]
      },
      {
        "name": "segregated",
        "routes": [
          {
            "destination": "1.0.4.0/32",
            "blackhole-route": true
          }
        ]
      },
      {
        "name": "shared",
        "routes": [{
          "destination": "1.0.2.0/32",
          "target-vpc": "Dev"
        }]
      },
      {
        "name": "standalone",
        "routes": [{
          "destination": "1.0.3.0/32",
          "target-vpn": {
            "name": "Perimeter_fw",
            "az": "b",
            "subnet": "Public"
          }
        }]
      }
    ]
  },
  {
    "name": "East",
    "asn": 64526,
    "region": "${GBL_REGION}",
    "features": {
      "DNS-support": true,
      "VPN-ECMP-support": true,
      "Default-route-table-association": false,
      "Default-route-table-propagation": false,
      "Auto-accept-sharing-attachments": true
    },
    "route-tables": ["core", "segregated", "shared", "standalone"],
    "tgw-attach": {
      "associate-to-tgw": "Main",
      "account": "shared-network",
      "region": "${HOME_REGION}",
      "tgw-rt-associate-local": ["core"],
      "tgw-rt-associate-remote": ["core"]
    },
    "tgw-routes": [
      {
        "name": "core",
        "routes": [
          {
            "destination": "1.1.0.0/32",
            "target-tgw": "Main"
          }
        ]
      },
      {
        "name": "segregated",
        "routes": [
          {
            "destination": "1.1.1.0/32",
            "target-tgw": "Main"
          }
        ]
      }
    ]
  }
]
```

- Creating a VPC Virtual Gateway

```
          "vgw": {
            "asn": 65522
          },
...
          "route-tables": [
            {
              "name": "Public_Shared",
              "routes": [
                {
                  "destination": "0.0.0.0/0",
                  "target": "IGW"
                }
              ]
            },
            {
              "name": "FWMgmt_azA",
              "routes": [
                {
                  "destination": "10.0.0.0/8",
                  "target": "VGW"
                }
              ]
            },
            {
              "name": "FWMgmt_azB",
              "routes": [
                {
                  "destination": "10.0.0.0/8",
                  "target": "VGW"
                }
              ]
            }
          ]
```

- Disable a Config rule on a per account basis - add this to either workload or mandatory accounts sections

```
      "aws-config": [
        {
          "regions": ["${HOME_REGION}", "${GBL_REGION}"],
          "excl-rules": ["ELB_LOGGING_ENABLED"]
        }
      ]
```

- Add SCP on a per account basis - add this to either workload or mandatory accounts sections

```
"scps": [
  "SCP 1",
  "SCP 2"
]
```

- Future description

```
{future sample}
```

- Future description

```
{future sample}
```

...Return to Customization Table of Contents