



30 lines (23 loc) · 1.49 KB

Preview

Code

Blame

Raw



Hardened Boundary Server

Task 1: Image Hardening

The goal of this task was to create a hardened Ubuntu 22.04 Image using Packer. Additionally, the provisioning agent was to be removed from the VM after the first boot.

- [Packer config](#)
- [OSCAP report before hardening](#)
- [OSCAP report after hardening](#)
- [Deployment screenshots](#)

Task 2: Boundary Installation

The scope of this task was to set up a boundary installation on the hardened VM, also deploying a web server and remote proxy. Additionally, firewall rules were to be configured. A boundary user was to be created with access to a single target.

On the client side, the boundary source code was to be adapted so the user would not have to provide an Auth Method ID.

- Client connection
 - The following picture shows how the authentication gets established.
 - However, we encountered an issue with curl which we could not resolve in time.

```

azureuser@09clientvm:~$ sudo /go/bin/boundary authenticate password --login-name=wp09user --password=file:///password.txt --scope-id=0_jycvEL1j0 -addr=https://boundary:8443
got auth-method from server: [ampw_pWnrCEPjH9]
authentication information:
  Account ID: acctpw_vQ05fdody
  Auth Method ID: ampw_pWnrCEPjH9
  Expiration Time: Mon, 30 Sep 2024 12:42:08 UTC
  User ID: u_32T0dmgIPj
Error: You must run:
  pass init your-gpg-id
before you may use the password store.
Error storing token in "pass" keyring: exit status 1
The token was not successfully saved to a system keyring. The token is:
at_7yTVD1wE4g_s18pSTcSHWaq4ekGpTdcKQbxjv9s2qldol37mymt3a3EywRNGpmParRQYDYr6GnZ3IF53z3FzFovThCdGzGRQHyteRKFmqhqbzu2Z9kceH2wuf4KUp11u5wn7z7NBPyW1k3KTS
It must be manually passed in via the BOUNDARY_TOKEN env var or -token flag. Storing the token can also be disabled via -keyring-type=none.
azureuser@09clientvm:~$ echo at_7yTVD1wE4g_s18pSTcSHWaq4ekGpTdcKQbxjv9s2qldol37mymt3a3EywRNGpmParRQYDYr6GnZ3IF53z3FzFovThCdGzGRQHyteRKFmqhqbzu2Z9kceH2wuf4KUp11u5wn7z7NBPyW1k3KTS > token.txt
azureuser@09clientvm:~$ sudo /go/bin/boundary connect http -addr=https://boundary:8443 -target-id=tcp_4tQecrVmo -token=file:///token.txt
curl: (35) Recv failure: Connection reset by peer
azureuser@09clientvm:~$

```

- Iptables

- The following picture shows the netfilter configuration of the VM hosting boundary which allows access only to the public services and SSH.

```

azureuser@wp09vm:~$ sudo iptables -L -v
Chain INPUT (policy DROP 101 packets, 8327 bytes)
 pkts bytes target     prot opt in     out     source            destination
 954K 247M ACCEPT     all  --  lo      any      anywhere          anywhere
 7849 859K ACCEPT     tcp  --  any     any      anywhere          anywhere          tcp dpt:ssh
 1480 280K ACCEPT     tcp  --  any     any      anywhere          anywhere          tcp dpt:8443
    0    0 ACCEPT     tcp  --  any     any      anywhere          anywhere          tcp dpt:9202

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source            destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source            destination
azureuser@wp09vm:~$

```

- Source code

- The boundary source code was modified on a client machine to statically fetch the Auth Method Id from the server
- The following picture shows the changes made to the `password.go` file

```

azureuser@09clientvm:~/boundary$ git diff
diff --git a/internal/cmd/commands/authenticate/password.go b/internal/cmd/commands/authenticate/password.go
index bcbad8bc2..74f1f230 100644
--- a/internal/cmd/commands/authenticate/password.go
+++ b/internal/cmd/commands/authenticate/password.go
@@ -6,8 +6,10 @@ package authenticate
 import (
     "errors"
     "fmt"
     "io"
     "os"
     "strings"
     "net/http"

     "github.com/hashicorp/boundary/api"
     "github.com/hashicorp/boundary/api/authmethods"
@@ -73,11 +75,20 @@ func (c *PasswordCommand) Flags() *base.Flags {
     // Usage: "The password associated with the login name. If blank, the command will prompt for the password to be entered interactively in a non-echoing way. Otherwise, this can refer to a file on disk (file:///) from which a password will be read or an env var (env://) from which the password will be read."
     )
     resp, err := http.Get("https://boundary:8443/custom/auth-method")
     if err != nil {
         fat.Println("Error fetching auth-method from server")
     }
     defer resp.Body.Close()
     body, err := io.ReadAll(resp.Body)
     authMethodIDDefault := strings.TrimSuffix(string(body), "\n")

     f.StringVar(&base.StringVar{
         Name: "auth-method-id",
         EnvVar: "BOUNDARY_AUTH_METHOD_ID",
         Target: &c.FlagAuthMethodId,
         Usage: "The auth-method resource to use for the operation.",
         Default: authMethodIDDefault,
     })
     if !c.parsedOpts.WithSkipScopeIDFlag {

```