

This repository

Search

Pull requests

Issues

Gist

gravitational / teleport

Watch 145

Star 3,895

Fork 180

<> Code

🔔 Issues 27

🔗 Pull requests 3

📁 Projects 0

📡 Pulse

📊 Graphs

re-introduce reverse tunnels into teleport #276

Merged klizhentas merged 2 commits into master from alexander/reversetunnel on Mar 20, 2016

💬 Conversation 19

📄 Commits 2

📄 Files changed 26

+818 -228



klizhentas commented on Mar 19, 2016

Contributor + 🗨️

Reverse tunnels are now first class citizens of teleport. There's no longer static configuration for reverse tunnel agents in the config. Instead, admins can add and remove reverse tunnels using `tctl reversetunnel` (hidden) commands.

- lists reverse tunnels

```
tctl reversetunnels ls
```

- updates or inserts reverse tunnel for 10 minutes

```
tctl reversetunnels upsert a.example.com 10.0.0.4:2023,10.0.0.5:2033 --ttl=10m
```

- delete a reverse tunnel

```
tctl reversetunnels del a.example.com
```

Teleport proxies watch changes in the reverse tunnels on the backend and spin up / spin down reverse tunnels according to these changes.

Reviewers

No reviews

Assignees

kontsevoy

Labels

enhancement

Projects

None yet

Milestone

No milestone

Notifications

🔔 Subscribe

You're not receiving notifications from this thread.

🔗 re-introduce reverse tunnels into teleport ...

✅ 6edd667

2 participants



🔗 klizhentas added the `enhancement` label on Mar 19, 2016

👤 kontsevoy was assigned by klizhentas on Mar 19, 2016



kontsevoy commented on Mar 19, 2016

Contributor + 🗨️

why do we need this in `tctl` at all? who would ever use this? shouldn't it be a library feature, not a tool feature?

And even if we need this for `tctl`, `reversetunnels` is a really bad name for a CLI command. "tunnel" is much friendlier. Shall we call it "clusters" maybe? To indicate this is how you connect clusters together?

<> kontsevoy commented on the diff on Mar 19, 2016

lib/defaults/defaults.go

View full changes

```
...    ...    @@ -89,6 +89,17 @@ const (
89    89        // DefaultReadHeadersTimeout is a default TCP timeout when we wait
90    90        // for the response headers to arrive
91    91        DefaultReadHeadersTimeout = time.Second
92    92        +
93    93        // ReverseTunnelsRefreshPeriod is a period for agents to refresh the
94    94        // state of the reverse tunnels (this will be removed once we roll
95    95        // events streams)
96    96        ReverseTunnelsRefreshPeriod = 3 * time.Second
```

```

97 +
98 + // ReverseTunnelAgentReconnectPeriod is the period between agent re
99 + ReverseTunnelAgentReconnectPeriod = 3 * time.Second
100 +
101 + // ReverseTunnelAgentHeartbeatPeriod is the period between agent hea
102 + ReverseTunnelAgentHeartbeatPeriod = 3 * time.Second

```



kontsevoy on Mar 19, 2016 Contributor

I think all these are too frequent for an SSH daemon with a tiny allowed CPU budget. Maybe it's my experience regularly playing with Raspberry Pi or Atom servers or tiny cloud VMs, but an SSH daemon really shouldn't spike to more than 1-3% CPU and plenty of servers out there have 10% of your MBP.

What would change if you set this to 30 seconds?



klizhentas on Mar 19, 2016 Contributor

this will go away anyways. Heartbeats are not CPU intensive they are simply sending packets over network



Reply...

<> **kontsevoy** commented on the diff on Mar 19, 2016

lib/reversetunnel/agent.go

[View full changes](#)

```

137 136             a.log.Infof("is closed, return")
138 137             return nil
139 -         default:
140 -         }
141 -         i++
142 -         if err = a.connect(); err != nil {
143 -             a.log.Infof("connect attempt %v: %v", i, err)
144 -             time.Sleep(time.Duration(min(i, 10)) * time.Second)
145 -             continue
138 +         case <-ticker.C:
139 +             if err = a.connect(); err != nil {
140 +                 a.log.Infof("connect attempt %v: %v", i, err)
141 +                 i++
142 +                 continue

```



kontsevoy on Mar 19, 2016 Contributor

1. what does "continue" do?
2. would be nice to back off further on reconnects, i.e. 1st attempt after 5 seconds, then 10, then 20... even one back-off step would be a big improvement, like `if i % 5 == 0 { continue }`



klizhentas on Mar 19, 2016 Contributor

constant time-reconnects solve a very particular problem here: in this case we need reconnect interval to be known to the other party that relies on it to detect dead connections:

<https://github.com/gravitational/teleport/blob/alexander/reversetunnel/lib/reversetunnel/srv.go#L609>



klizhentas on Mar 19, 2016 Contributor

re: continue - you are right, continue is not necessary here. I will remove it



Reply...

<> **kontsevoy** commented on the diff on Mar 19, 2016

lib/reversetunnel/agentpool.go

[View full changes](#)

```

127 +
128 +     for _, key := range agentsToAdd {

```

```

129 +         m.Infof("adding %v", &key)
130 +         agent, err := NewAgent(key.addr, key.domainName, m.cfg.Hosts)
131 +         if err != nil {
132 +             return trace.Wrap(err)
133 +         }
134 +         go func() {
135 +             if err := agent.Start(); err != nil {
136 +                 m.Warningf("%v failed to start", agent)
137 +             }
138 +         }()
139 +         m.agents[key] = agent
140 +     }
141 +     return nil
142 + }

```

**kontsevoy** on Mar 19, 2016 Contributor

gut feeling: this function (and its descendants) does more looping, parsing and memory allocations than it probably should. definitely a room for a better algo here.

**klizhentas** on Mar 19, 2016 Contributor

well, I assure you this won't be the slowest part of teleport. we are talking about having 1-2 tunnels at max per cluster, so this function will do nothing most of the time.

**klizhentas** on Mar 19, 2016 Contributor

I can profile it though to see if it's any trouble in terms of efficiency.

**kontsevoy** on Mar 19, 2016 Contributor

i wouldnt worry about profiling/performance, but I'd see if a much simpler algo will do. less code, more readability, etc. it just does *a lot* at a first glance



Reply...

<> **kontsevoy** commented on the diff on Mar 19, 2016

tool/tctl/main.go

[View full changes](#)

```

...   ...   @@ -116,6 +124,19 @@ func main() {
116   124         authServers := app.Command("authservers", "Operations with user and
117   125         authServerAdd := authServers.Command("add", "Add a new auth server
118   126
127   +         // operations with reverse tunnels
128   +         reverseTunnels := app.Command("reversetunnels", "Operations with re
129   +         reverseTunnelsList := reverseTunnels.Command("ls", "List reverse tu
130   +         reverseTunnelsDelete := reverseTunnels.Command("del", "Deletes reve
131   +         reverseTunnelsDelete.Arg("domain", "Comma-separated list of reverse
132   +             Required().StringVar(&cmdReverseTunnel.domainNames)
133   +         reverseTunnelsUpsert := reverseTunnels.Command("upsert", "Update or
134   +         reverseTunnelsUpsert.Arg("domain", "Domain name of the reverse tunne

```

**kontsevoy** on Mar 19, 2016 Contributor

we've been slowly phasing out word "domain" from Teleport. We have node names and node GUIDS. Do reverse tunnels introduce sites? Should we have site names? What is a "domain"?

**klizhentas** on Mar 19, 2016 Contributor

domain is a certificate authority domain - it's still there.

**klizhentas** on Mar 19, 2016 Contributor

node GUIDs is a different thing though



Reply...

<> **kontsevoy** commented on the diff on Mar 19, 2016

tool/teleport/main.go

View full changes

... @@ -104,6 +108,12 @@ func run(cmdlineArgs []string, testRun bool) (executed

104 108

105 109



106 110

111 +

switch command {


case start.FullCommand():

if ccf.HTTPProfileEndpoint {

-  **kontsevoy** on Mar 19, 2016 Contributor
- sweet
-  **klizhentas** on Mar 19, 2016 Contributor
- yep it's a helpful one


 Reply...



kontsevoy commented on Mar 19, 2016 Contributor + 

k, looks good, although I made a few comments in the diffs, particularly about frequency and efficiency of tunnel bookkeeping overhead.



klizhentas commented on Mar 19, 2016 Contributor + 


regarding naming. I chose the name reverse tunnel because it's a wide spread term for this sort of thing:
<http://unix.stackexchange.com/questions/46235/how-does-reverse-ssh-tunneling-work>

So I would like to avoid inventing new terminology if the existing one will work. We can come up with some shortcuts for tctl to reduce typing, e.g. `rtunnels` or `revtunnels`

Reverse SSH tunnels are quite wide-spread practice, so I think it makes sense to expose it via tctl. E.g. gravity will be using this feature. In addition to that you may temporary create a a tunnel to remote organization for debugging purposes.

I made it a hidden feature though - so admins will know that they need it before we confuse them :)



kontsevoy commented on Mar 19, 2016 Contributor + 


k, merge. I'd prefer `rts` to `reversetunnels` . I like reading but I don't like typing.



klizhentas commented on Mar 19, 2016 Contributor + 

k, I will rename to `rts`

  shorten command name ✓ 3ede574

 **klizhentas** merged commit `6456f0a` into `master` on Mar 20, 2016 View details

1 check passed

 **klizhentas** deleted the `alexander/reversetunnel` branch on Mar 20, 2016



Write Preview


AA B i “ < > ↺ ⋮ ≡ ✓ ↶ @ *

Leave a comment

Attach files by dragging & dropping, [selecting them](#), or pasting from the clipboard.

 Styling with Markdown is supported

Comment

 **ProTip!** Add `.patch` or `.diff` to the end of URLs for Git's plaintext views.

