Non-interactive AWS Azure Login

Eric Bailey

20th December, 2019[1]

Password management

For password management, use pass[2].

⟨*Get the password from the store* 1⟩≡ (4)

```
pass show "Azure AD" 2>/dev/null | head -n1
```

This assumes `Azure AD` is in the password store.

On top of that, use the pass-otp[3] extension to manange OTP tokens.

⟨*Generate an OTP token* 2⟩≡ (5)

```
pass otp show "Azure AD"
```

This assumes `Azure AD` is configured.

With these two tools, automate refreshing AWS credentials using Azure AD.

Automated interaction

Spawn a non-interactive aws-azure-login[4] process, forcing a credential refresh, even if they are still valid.

⟨*Spawn aws-azure-login process* 3⟩≡ (6)

```
spawn -noecho aws-azure-login -force-refresh -no-prompt
```

This requires the environment variable, `AWS_PROFILE`, to be set, and `~/.aws/config` to be properly configured.

Alternatively, send the username...

```
expect "*Username*" {
    set username [ exec pass show "Azure AD" | awk "/Username:/ { print \$2 }" ]
    send "$username\n"
}
```

... and duration to their respective prompts.

```
expect "*Duration*" { send "8\n" }
```

The main point of this script is to obviate the need to manually provide a password and OTP token, letting Expect[5] `interact` on the user's behalf.

When prompted for the `password`, get it from the store and `send` it.

⟨*Send the password when expected* 4⟩≡ (6)

```
expect "*Password:*" {
    set password [ exec ⟨Get the password from the store 1⟩ ]
    send "$password\r"
}
```

When prompted for an OTP token, generate one and `send` it.

⟨*Send the OTP token when expected* 5⟩≡ (6)

```
expect "*Verification Code:*" {
    set verification_code [ exec ⟨Generate an OTP token 2⟩ 2>/dev/null ]
    send "$verification_code\r"
}
```

⟨* 6⟩≡

```
#! /usr/bin/env expect

⟨Spawn aws-azure-login process 3⟩

⟨Send the password when expected 4⟩

⟨Send the OTP token when expected 5⟩

interact
```

[1] Updated 06.06.2021 04:22 GMT.

[2] https://www.passwordstore.org/

[3] https://github.com/tadfisher/pass-otp

[4] https://github.com/sportradar/aws-azure-login

[5] https://core.tcl-lang.org/expect/index