T&I Incubator: OIDC Support for SSH under Windows

Final demo – 21st September 2021

Dmytro Dehtyarov

www.geant.org

Agenda

- Motivation & Background
- Port oidc-agent to Windows
- Communication with oidc-agent
- Integration with putty
- Outlook

Agenda

- Motivation & Background
- Port oidc-agent to Windows
- Communication with oidc-agent
- Integration with putty
- Outlook

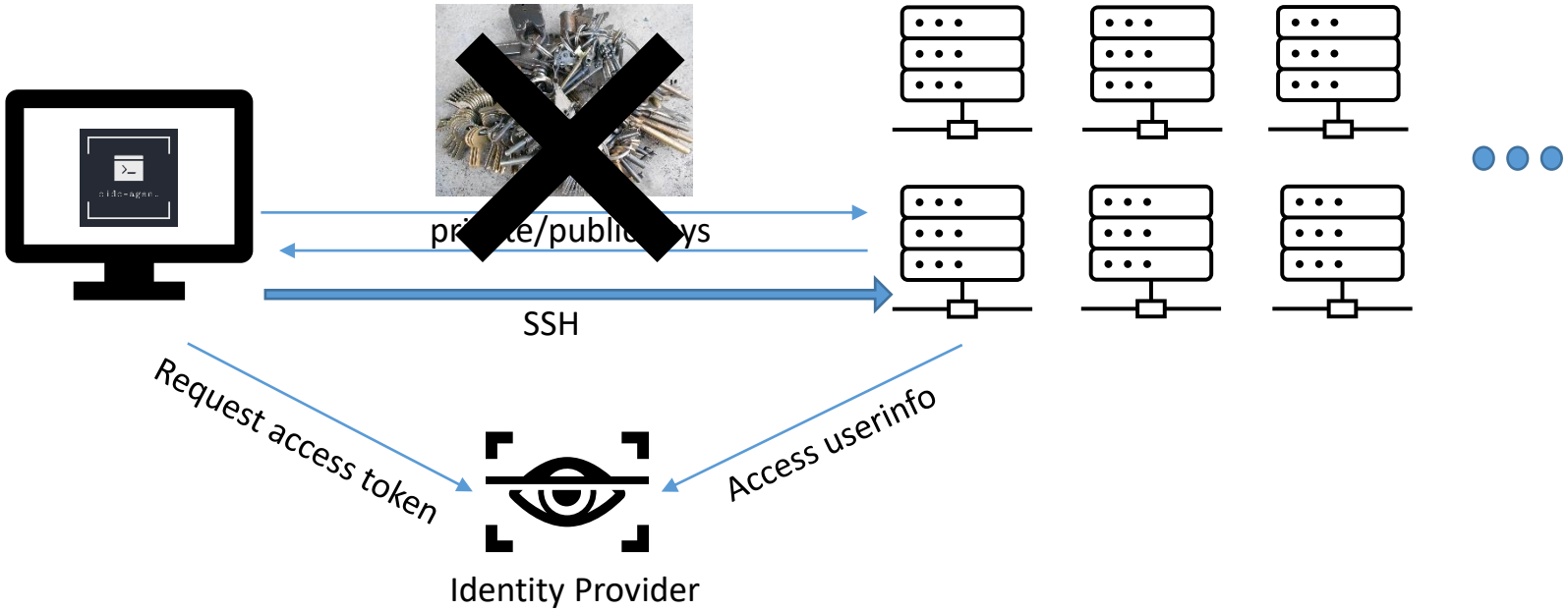
Motivation

- SSH Key Management
- Risks of untracked and unmanaged SSH Keys
- 90% of keys are no longer used [1]

[2]

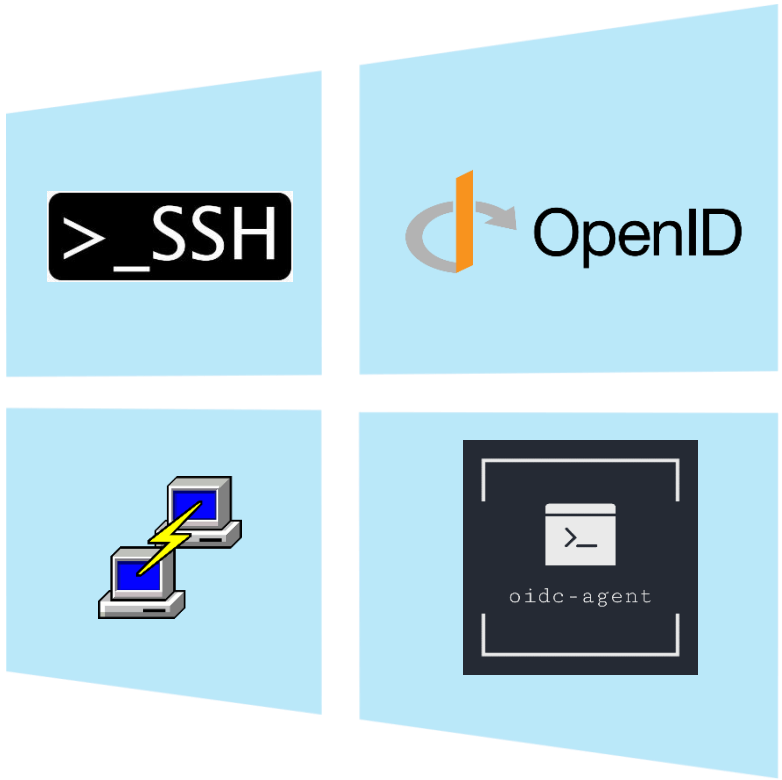
Motivation

Access Tokens!



Goals

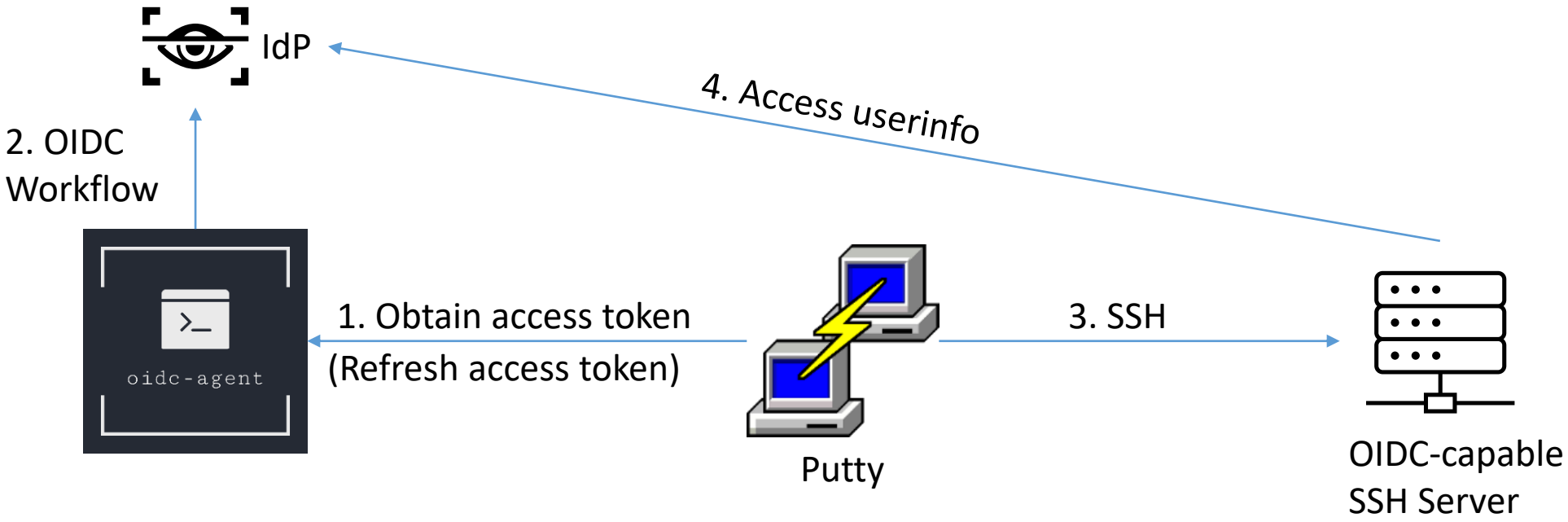
- High-Level: Integrate OIDC with SSH (client-side) under **Windows**
  - Port oidc-agent to Windows
  - Integrate oidc-agent into SSH client Putty
  - Extend putty for access token based authentication



[3][4][5][6][7]

High-Level Architecture

- oidc-agent VS. ssh-agent



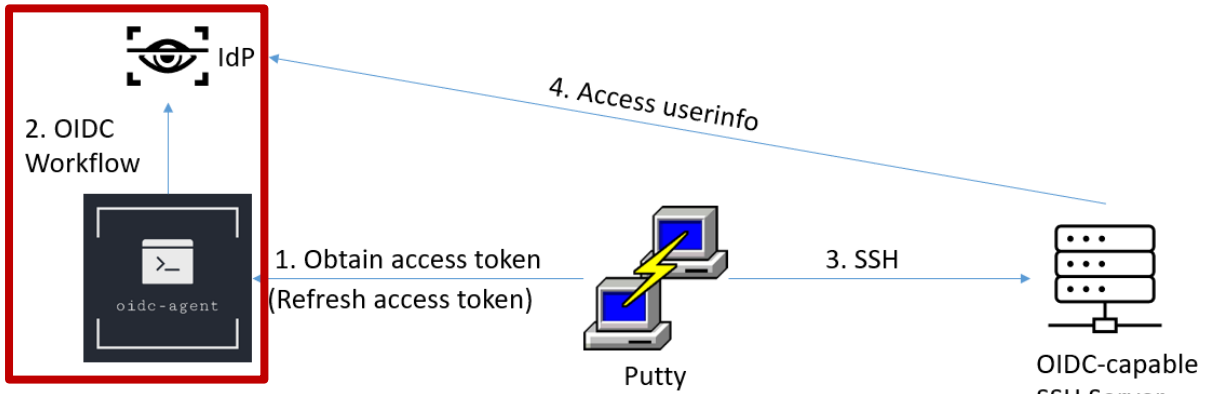
Agenda

- Motivation & Background
- Port oidc-agent to Windows
- Communication with oidc-agent
- Integration with putty
- Outlook

oidc-agent

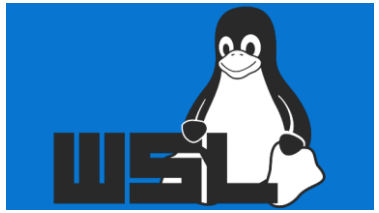
- Command-line tools
  - oidc-agent
  - oidc-gen
  - oidc-add
  - oidc-token
- Available for Linux and MacOS

https://github.com/indigo-dc/oidc-agent



Possibilities to port oidc-agent

- ✗ Windows Subsystem for Linux 2
  - No changes to source code
  - Not user-friendly
- ✗ Build natively
  - Rewrite platform-dependent code (IPC, pthreads, etc)
- ✓ POSIX emulation layer – Cygwin, MSYS
  - Moderate changes to source code
  - Semi-native approach



[8]

libsodium [9]

Libmicrohttpd [10]

[11]

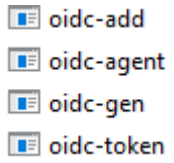
[12]

Build oidc-agent with MSYS2

- Source code modifications
  - Using compiler directives
  - Excluded functionalities
  - Implemented registry connector
  - Adapted building procedures
- Build with ‚make‘
  - Same codebase
  - Produces .exe binaries
- Build&Installation script

```
#ifdef __MSYS__
#include <windows.h>
#endif
```

```
GEANT@finrod2 MSYS ~/oidc-agent
$ make
Compiled src/utils/registryConnector.c successfully!
Compiled src/ipc/cryptCommunicator.c successfully!
Compiled src/ipc/ipc.c successfully!
Compiled src/api/comm.c successfully!
Compiled src/api/tokens.c successfully!
Linking bin/oidc-agent complete!
Linking bin/oidc-gen complete!
Linking bin/oidc-add complete!
Compiled src/api/comm.c with pic successfully!
Compiled src/api/tokens.c with pic successfully!
Compiled src/ipc/ipc.c with pic successfully!
Compiled src/ipc/cryptCommunicator.c with pic successfully!
Compiled src/utils/registryConnector.c with pic successfully!
Compiled src/oidc-token/oidc-token.c successfully!
Linking bin/oidc-token complete!
```

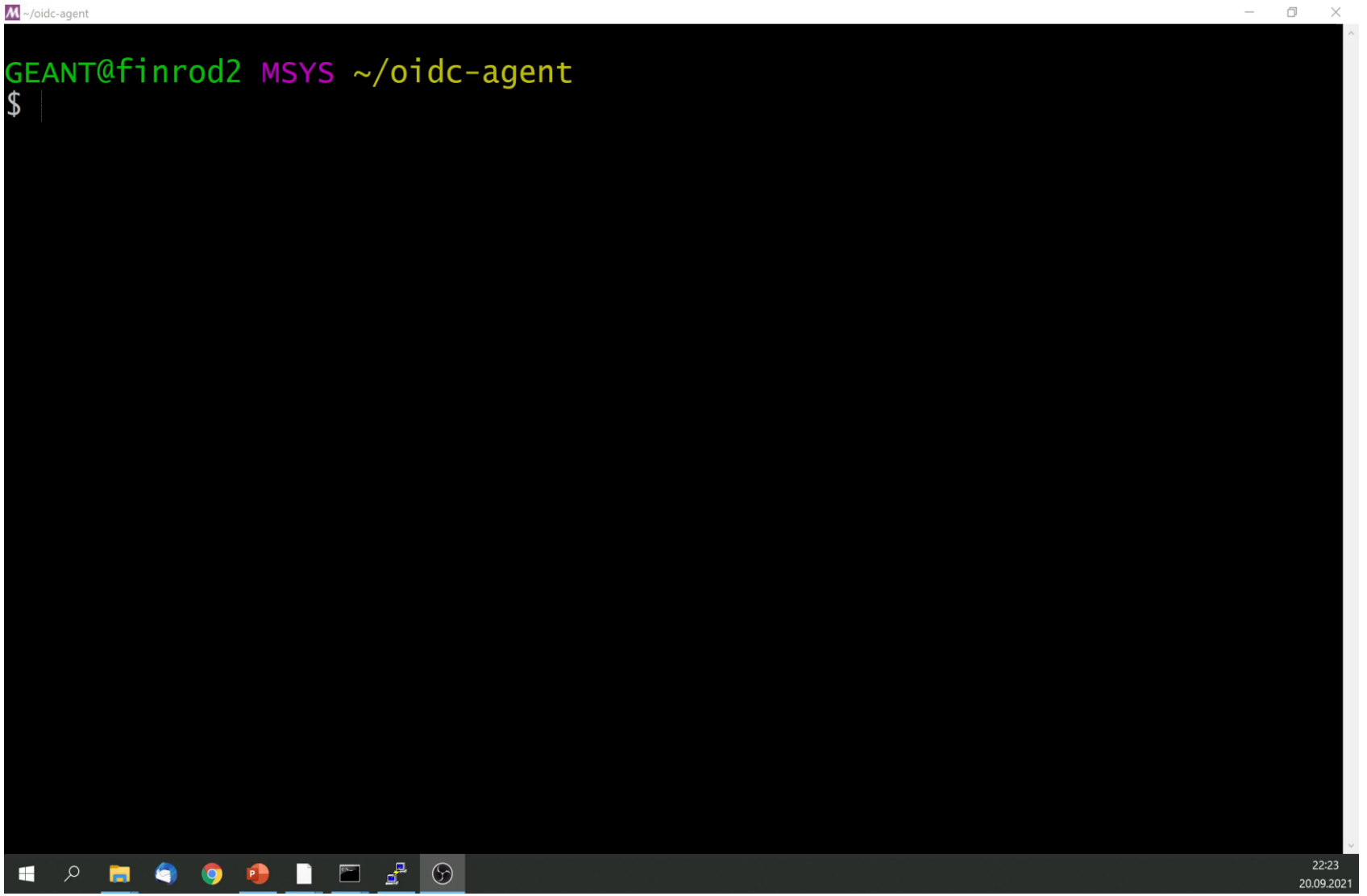


oidc-agent on Windows

- Github
  - https://github.com/indigo-dc/oidc-agent/tree/windows

Demo

oidc-agent on Windows

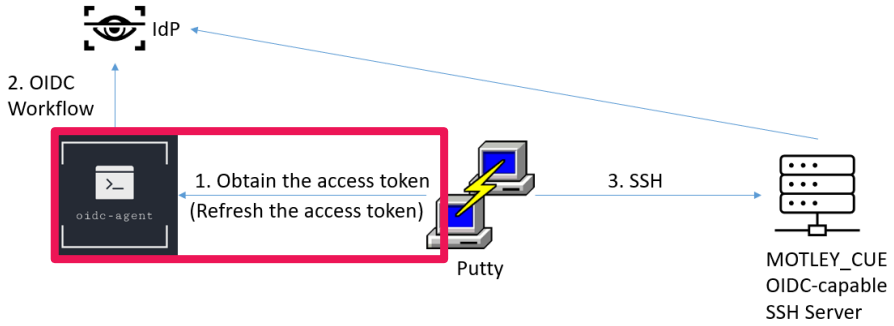


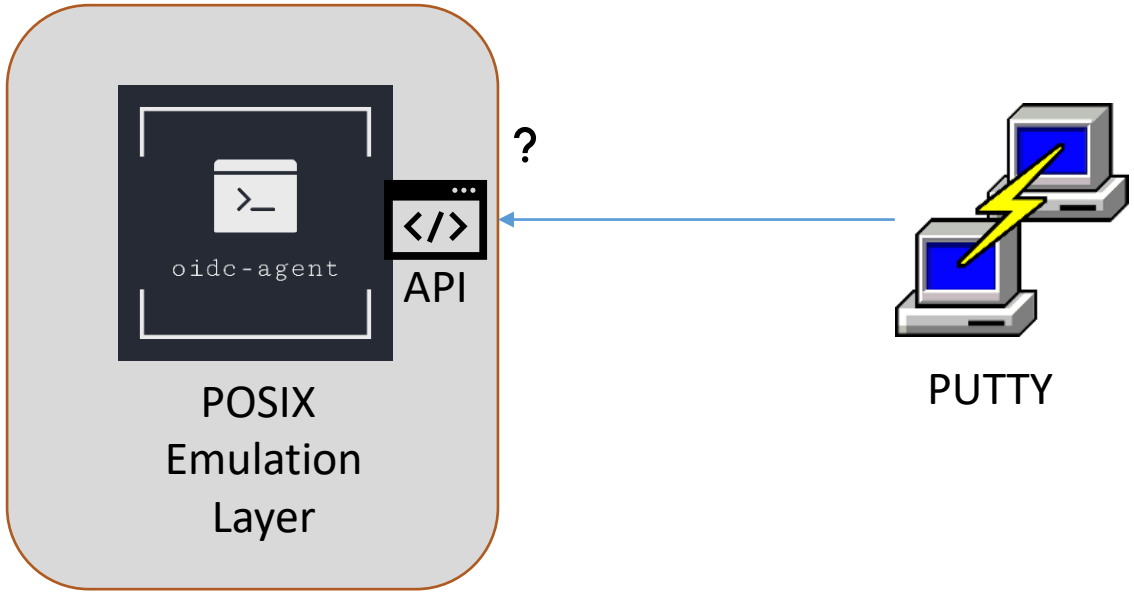
Agenda

- Motivation & Background
- Port oidc-agent to Windows
- Communication with oidc-agent
- Integration with putty
- Outlook

liboidc-agent API

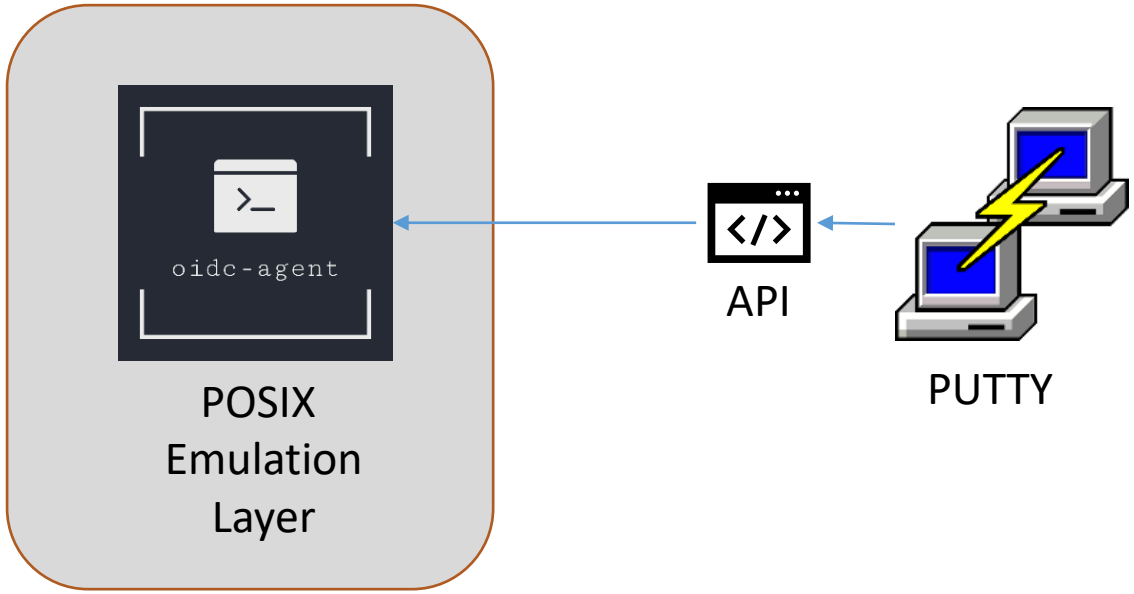
- Static/dynamic library
  - Unix Domain Sockets
  - Requires compatibility layer!
- Goal: compile API library natively





Native liboidc-agent API library

- Reimplemented IPC using winsock
- Talk directly with emulated sockets
  - Authentication handshake
- Compiled using mingw-w64 (MSYS2 or cross-compile on Linux)



Native liboidc-agent API library

- Originally only for oidc-token
- Refactored & extended for other tools
- Following endpoints available:
  - getAccessToken
  - getAccessTokenForIssuer
  - getLoadedAccountsList

Agenda

- Motivation & Background
- Port oidc-agent to Windows
- Communication with oidc-agent
- Integration with putty
- Outlook

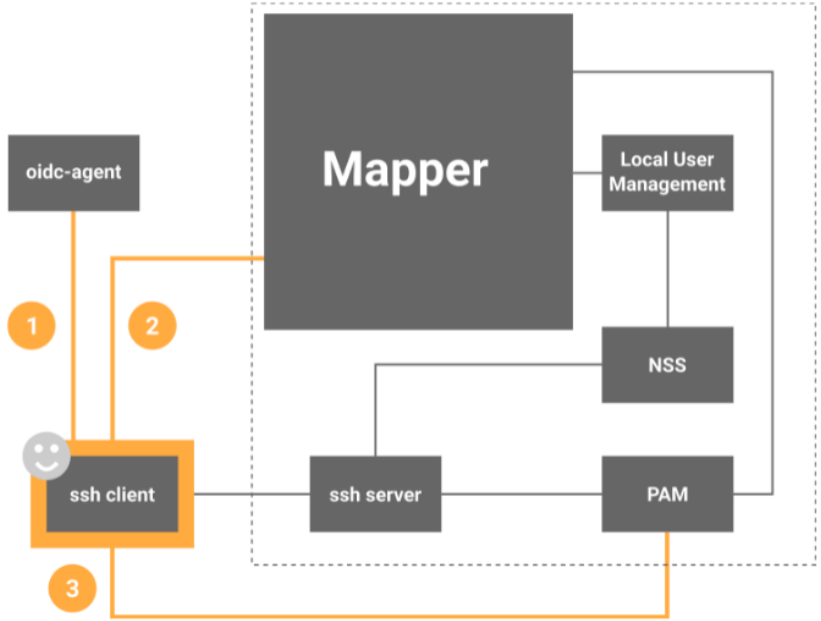
Putty

- Client for **SSH**, Telnet, …
- Putty, Plink, PSFTP, PSCP
- Originally written for Windows
- Supports authentication:
  - no-auth
  - password
  - public key
  - Gssapi
  - **keyboard-interactive**
  - **… OIDC access tokens**

OIDC Based Authentication

- Server:
  - SSH server with pam-ssh-oidc
  - Mapping daemon motley_cue

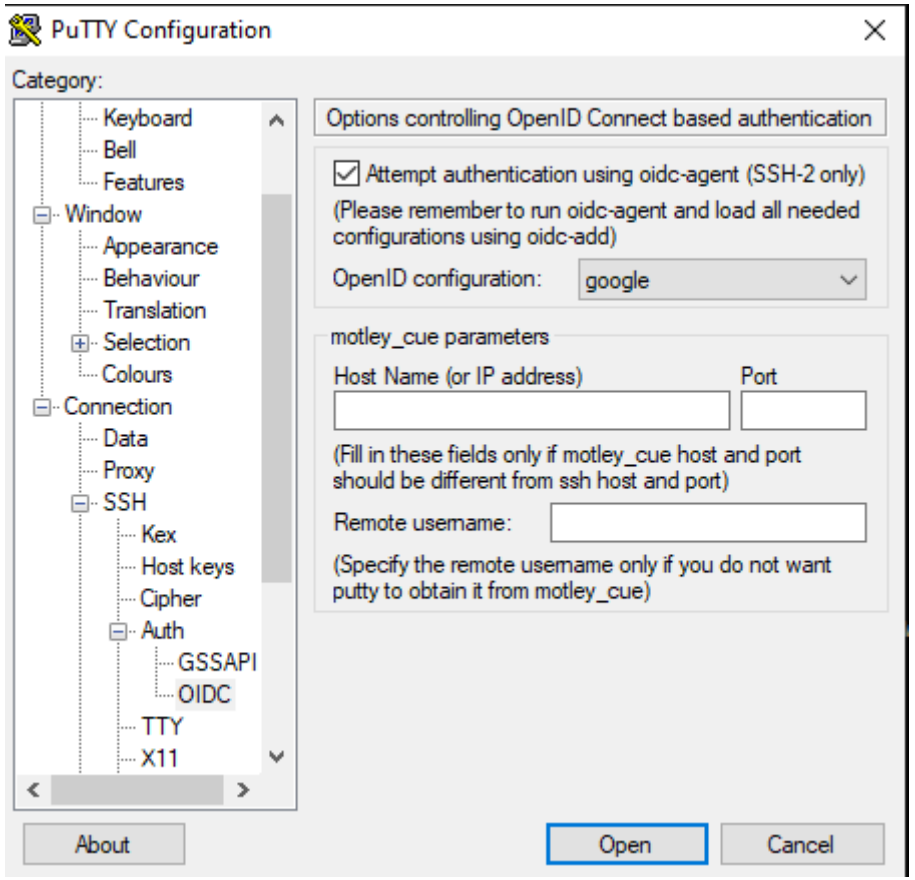
1. Retrieve access token
2. Deploy local account/get username
3. Input access token when prompted
   - Subtype of keyboard-interactive auth



[13]

OIDC Based Authentication

- Available for:
  - Putty
  - Plink
  - PSFTP
  - PSCP
- Built using mingw-w64
  - Windows (MSYS2 – mingw32)
  - Cross-compile on Linux

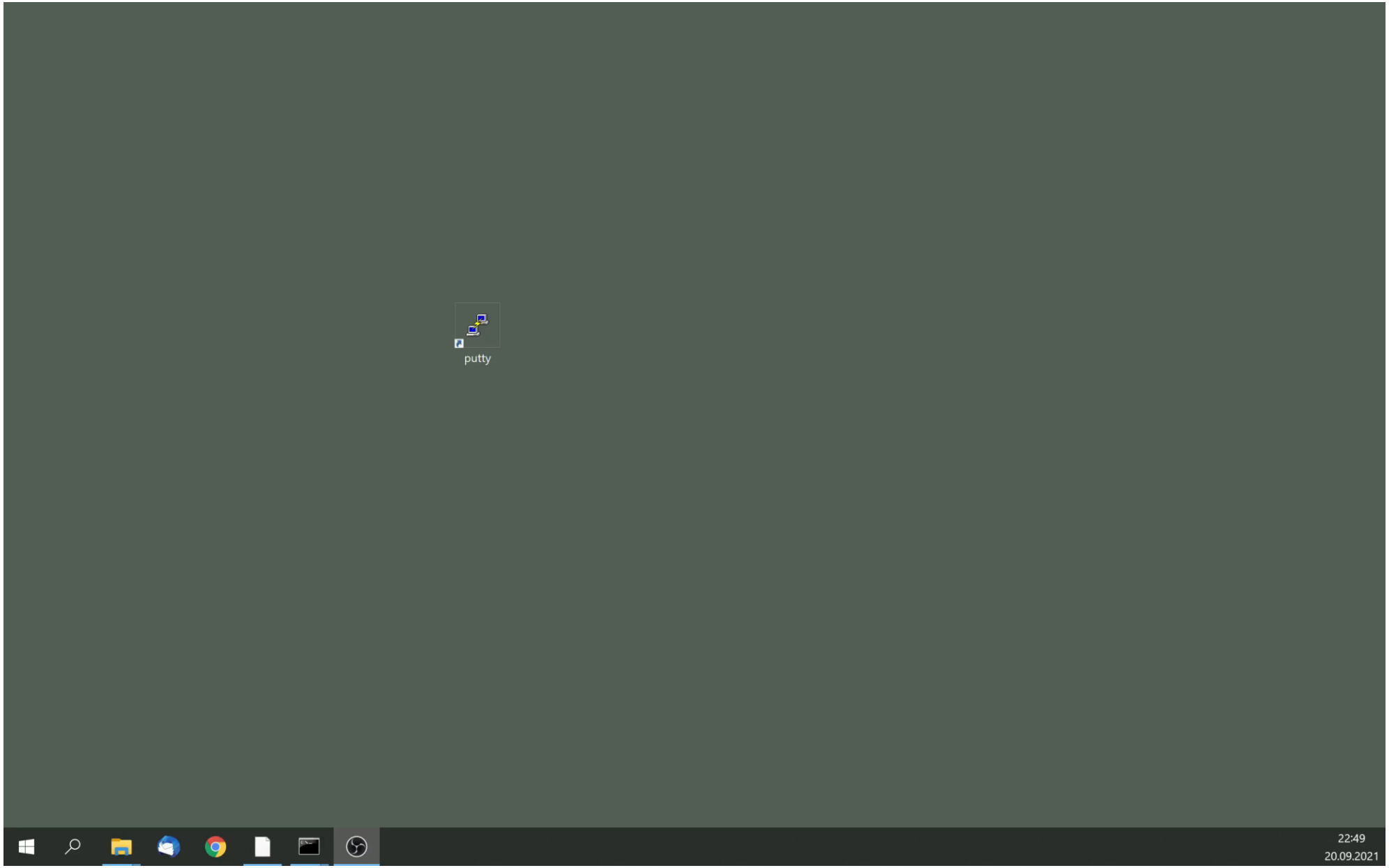



```
C:\msys64_putty\home\GEANT\putty\windows>plink -oidc-agent google -mc-host ssh-oidc-demo.data.kit.edu ssh-oidc-demo.data.kit.edu
-- Keyboard-interactive authentication prompts from server: ------------------
-- Passed OIDC access token ------------------------------------------------
-- End of keyboard-interactive prompts from server --------------------------
Access granted. Press Return to begin session.
```

OIDC Based Authentication

- Git:
  - https://git.scc.kit.edu/m-team/putty/-/tree/oidc-support

Demo


putty
22:49
20.09.2021

Agenda

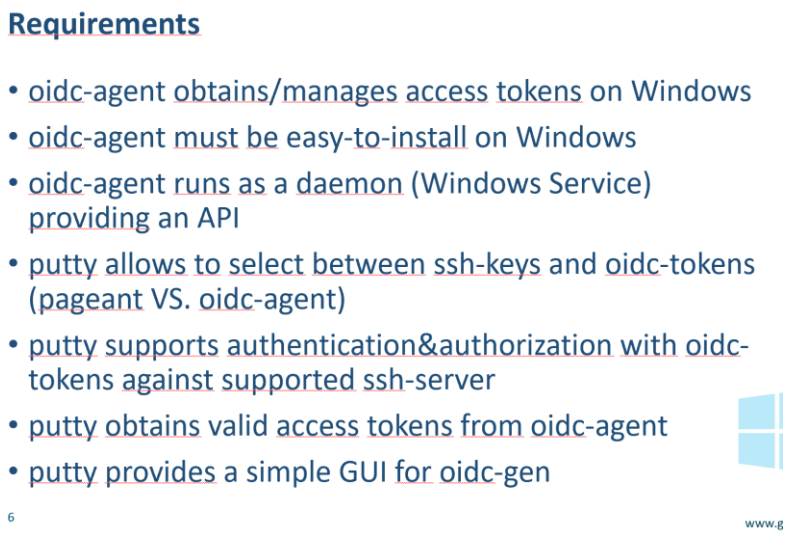
- Motivation & Background
- Port oidc-agent to Windows
- Communication with oidc-agent
- Integration with putty
- Outlook

Outlook

- Provide installer for oidc-agent
  - Better user-experience
- Build liboidc-agent.lib for Windows environment (MSVC)
- Provide OIDC support for Putty under supported compilers
  - Clang
  - MSVC
  - LCC
- Discuss and publish putty extension

Summary

- All requirements satisfied
- oidc-agent runs on Windows
- liboidc-agent works on Windows
- Putty supports authentication with access tokens for SSH



Thank you

Any questions?

www.geant.org

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 856726 (GN4-3).

References

[1] https://www.ssh.com/academy/iam/ssh-key-management

[2] https://images.app.goo.gl/8CTV52CRGRxJrBRLA

[3] https://images.app.goo.gl/jzgLw2zjqKzwi7cp9

[4] https://images.app.goo.gl/tKo77xxstdu1DUJr5

[5] https://images.app.goo.gl/bUgQzq5YaTPamEXP8

[6] https://github.com/indigo-dc/oidc-agent/blob/master/logo.png

[7] https://images.app.goo.gl/TJy4y5WigecQJ3sU7

[8] https://docs.microsoft.com/en-us/windows/wsl/

[9] https://images.app.goo.gl/jzXenF4tMquP19HCA

[10] https://images.app.goo.gl/KTBckfa6R4mibR8z9

[11] https://images.app.goo.gl/CBiCK5SqUDRksqt69

[12] https://images.app.goo.gl/M1eHHZSzY1cc9X37A

[13] https://docs.google.com/presentation/d/17HM11YjafC5VA4_o2EjNrtbRqJGgQP0q92C_uqFAM6A/edit

[14] https://images.app.goo.gl/Jaxk9RcesMzGCcS99