



東京大学  
THE UNIVERSITY OF TOKYO



東京大学情報基盤センター  
INFORMATION TECHNOLOGY CENTER, THE UNIVERSITY OF TOKYO

# オンライン講習会の手引き 「MATLABの実行方法」

東京大学情報基盤センター

質問は中島まで:[nakajima\(at\)cc.u-tokyo.ac.jp](mailto:nakajima(at)cc.u-tokyo.ac.jp)

(相談窓口にはメールしないでください)

# 概要

- 講習会ホームページ
  - <https://www.cc.u-tokyo.ac.jp/events/lectures/177/>
- 本講習会では、理解を深めるため、[Wisteria/BDEC-01スーパーコンピューターシステム \(Aquarius\)](#) を利用した実習を実施する予定である。
- 本講習会はZoomによるオンライン講習会として実施される予定である。
- 本資料は、オンライン講習会受講のための準備について記載したものである。

# お願い等

- 東京大学情報基盤センタースーパーコンピュータシステム利用  
規程を熟読し、遵守してください。
- ハンズオンのためのPC, Zoom及びスパコンへ接続するためのネットワーク環境は各受講者でご準備ください。
- PCは Windows/Microsoft Update, Apple Security Updateなどで最新のセキュリティアップデートを行ってください。
- 必ずウィルス対策ソフトウェアをインストールし、ウィルス検索を実行して問題がないことを事前に確認してから受講してください。
  - セキュリティ対策未実施の場合はオンライン講習会受講を認めません。
- OSは、Windows、Macどちらでも構いませんが、SSHを用いてスーパーコンピュータへ接続ができることが必須です(後述)。
- 受講生にセンターのスーパーコンピュータを1月間利用できる無料アカウント(お試しアカウント)を発行します。

# 準備について

- 以下に示す「PC上でのソフトウェアの準備」について受講日までに完了していることを前提として、講習会を実施いたしますので、よろしくお願いいたします。
  - 「スパコンへのログイン」以降は当日実施します

- PC上のソフトウェア類の準備
- スパコンへのログイン
- ログインしたら・・・

# Windows向け環境

- **WSL (Windows Subsystem for Linux)**
  - 簡単・お勧め
  - Windows 10以降
  - 「管理者権限」が必要
- **Cygwin**
  - 以前はお勧めしていたが、インストールがやや煩雑、時間もかかる
  - メモリ・ディスクスペースもそれなりに必要
  - 管理者権限不要
- **SSHログインのみであれば、Windows PowerShellもしくはWindowsコマンドプロンプトからも可能**
  - 管理者権限不要
  - OpenSSH Clientがインストールされていることを確認
    - 下記に従って確認, インストール (OpenSSH Serverは不要)
    - [https://docs.microsoft.com/ja-jp/windows-server/administration/openssh/openssh\\_install\\_firstuse](https://docs.microsoft.com/ja-jp/windows-server/administration/openssh/openssh_install_firstuse)

# Cygwin: Windows上のUNIXライクな環境

<https://www.cygwin.com/>

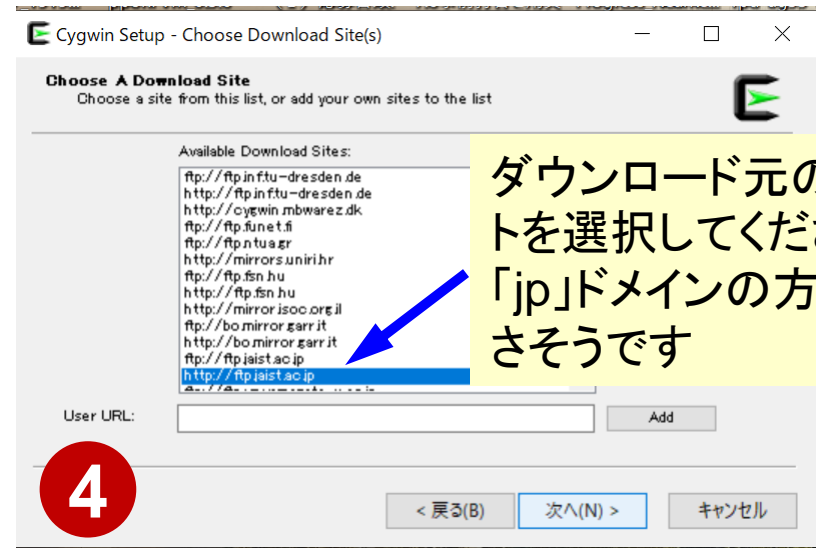
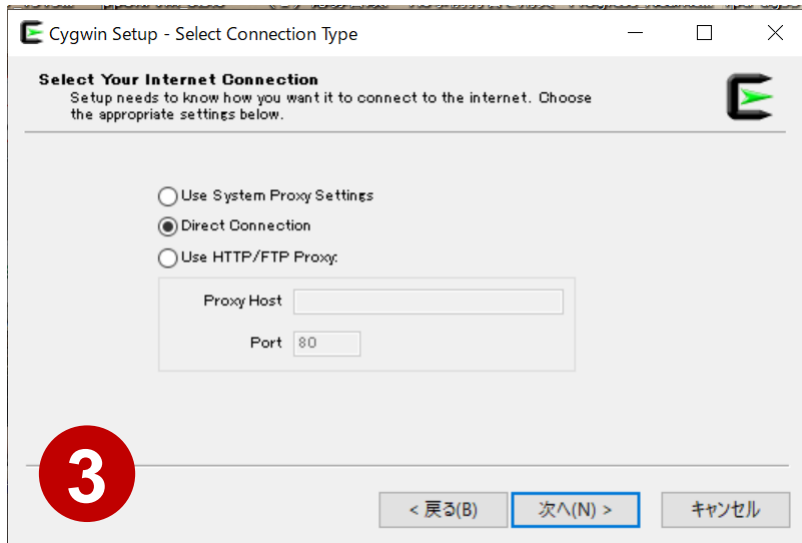
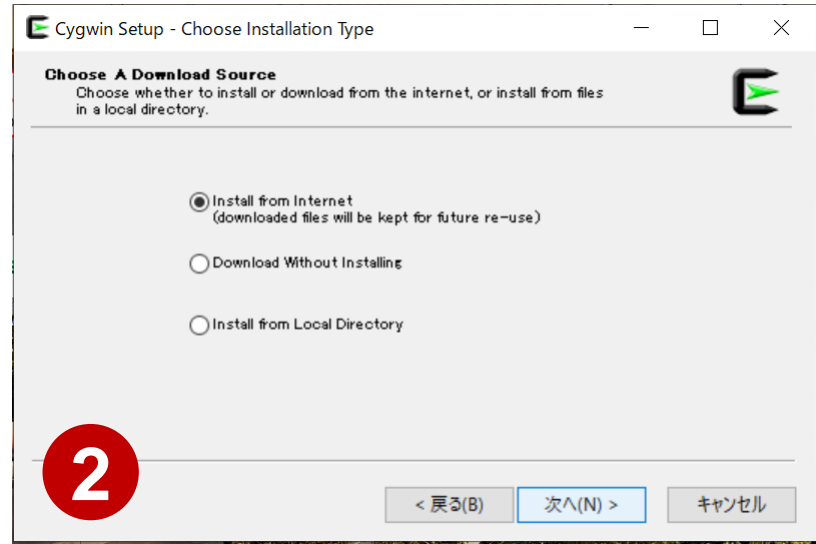
- 利用者ガイド
  - <https://cygwin.com/cygwin-ug-net.html>
- まずやるべきこと
  - インストーラ (setup-x86\_64/32.exe) を予め入手
  - インターネットに接続し、インストーラをダブルクリック
  - 以下指示に従ってインストールを進めてください

以下しばらくはCygwinの話



# 指示に従ってください

<https://www.cygwin.com/>





# まずはデフォルト機能のインストールから

<https://www.cygwin.com/>

- 基本的な機能はデフォルトのインストールでOKですが、本講習会で必要なものが抜けている可能性があります。
  - 従ってマニュアルでインストールする必要があります
  - インストーラをダブルクリックすれば後で追加も可能です
- **本講習会では下記が必須です（デフォルトのインストールでは抜けている可能性あり）**
  - **gcc-core (for C/C++ users)**
  - **gcc-fortran (for Fortran users)**
  - **openssh (for all users)**
  - **openssl (for all users)**
  - **make (for all users)**
  - **emacs, vim etc.**
- **インストールされているかどうかは確認が可能です**

# “gcc-core”の有無に関するチェック

Type “gcc-core”

Cygwin Setup - Select Packages

Select packages to install

View Full Search gcc-core Clear

Keep  Best  Sync  Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-core		Skip	<input type="checkbox"/>	Devel	16,464k	GCC for Cygwin 32bit toolchain (C, OpenMP)
djgpp-gcc-core		Skip	<input type="checkbox"/>	Devel	7,926k	GCC for DJGPP toolchain (C)
gcc-core	9.3.0-1	Keep	<input type="checkbox"/>	Devel	20,500k	GNU Compiler Collection (C, OpenMP)
mingw64-i686-gcc-core		Skip	<input type="checkbox"/>	Devel	16,851k	GCC for Win32 (i686-w64-mingw32) toolchain (C, OpenMP)
mingw64-x86_64-gcc-core		Skip	<input type="checkbox"/>	Devel	17,464k	GCC for Win64 toolchain (C, OpenMP)

Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

20:19 2020/04/14

# “gcc-core”の有無に関するチェック

Type “gcc-core”

Select Packages  
Select packages to install

View Full Search gcc-core Clear

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-core		Skip	<input type="checkbox"/>	Devel	16,464k	GCC for Cygwin 32bit toolchain (C, OpenMP)
gcc-core	9.3.0-1	Keep	<input type="checkbox"/>	Devel	7,926k	GCC for DJGPP toolchain (C)
gcc-core		Skip	<input type="checkbox"/>	Devel	20,500k	GNU Compiler Collection (C, OpenMP)
gcc-core		Skip	<input type="checkbox"/>	Devel	16,851k	GCC for Win32 (686-w64-mingw32) toolchain (C, OpenMP)
mingw64-x86_64-gcc-core		Skip	<input type="checkbox"/>	Devel	17,464k	GCC for Win64 toolchain (C, OpenMP)

“Keep”と出てきたら  
“gcc-core”はインストール済み

< 戻る(B) 次へ(N) > キャンセル

Hide obsolete packages

20:19  
2020/04/14

# “g++”のインストール例 (1/4)

Type “g++”

Cygwin Setup - Select Packages

Select Packages  
Select packages to install

View Full Search  Clear

Keep  Best  Sync  Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	<input type="checkbox"/>	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
dragon-gcc-g++		Skip	<input type="checkbox"/>	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		Skip	<input type="checkbox"/>	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,781k	GCC for Win64 toolchain (C++)

“Skip” が現れたら「未インストール」を意味する

以下「g++」を例にマニュアルインストールの実施方法を紹介する

Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

# “g++”のインストール例 (2/4)

Cygwin Setup - Select Packages

Select Packages  
Select packages to install

View Full Search g++ Clear  Keep  Best  Sync  Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	<input type="checkbox"/>	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
dragon-gcc-g++		Skip	<input type="checkbox"/>	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		Skip	<input type="checkbox"/>	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,781k	GCC for Win64 toolchain (C++)

プルダウンメニュー

Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

20:18  
2020/04/14

# “g++”のインストール例 (3/4)

Cygwin Setup - Select Packages

Select Packages  
Select packages to install

View Full Search g++ Clear  Keep  Best  Sync  Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	<input type="checkbox"/>	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
diipp-gcc-g++		Skip	<input type="checkbox"/>	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		Skip	<input checked="" type="checkbox"/>	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,781k	GCC for Win64 toolchain (C++)

Uninstall  
 Skip  
 7.4.0-1  
 8.3.0-1 (Test)  
 9.2.0-1 (Test)  
 9.2.0-2  
 9.2.0-3  
 9.3.0-1

適切なバージョンを選択  
(通常は最新版)

Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

20:18 2020/04/14

# “g++”のインストール例 (4/4)

Cygwin Setup - Select Packages

Select Packages  
Select packages to install

View Full Search g++ Clear  Keep  Best  Sync  Test

Package	Current	New	Src?	Categories	Size	Description
cygwin32-gcc-g++		Skip	<input type="checkbox"/>	Devel	10,456k	GCC for Cygwin 32bit toolchain (C++)
diipp-gcc-g++		Skip	<input type="checkbox"/>	Devel	8,279k	GCC for DJGPP toolchain (C++)
gcc-g++		9.3.0-1	<input type="checkbox"/>	Devel	16,257k	GNU Compiler Collection (C++)
mingw64-i686-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,358k	GCC for Win32 (i686-w64-mingw32) toolchain (C++)
mingw64-x86_64-gcc-g++		Skip	<input type="checkbox"/>	Devel	14,781k	GCC for Win64 toolchain (C++)

「Skip」のかわりにバージョン番号が出てきたら“g++”のインストール準備完了  
(インストールは完了していない)

ここをクリック

Hide obsolete packages

< 戻る(B) 次へ(N) > キャンセル

21:58  
2020/04/14

# WSL (1/2)

## Windows-Subsystem-for-Linux

- Windows 10 Ver.2004 (build 19041) or later, Windows 11
  - Win+r > type [winver] > click [OK]
- 「管理者権限で」 PowershellまたはWindowsコマンドプロンプトを立ち上げ, 下記を入力すれば一通りインストールしてくれる
  - ssh-keygen, cc, gcc, nanoなどはデフォルトでインストール

```
$ wsl --install
```



# WSL (2/2)

## Windows-Subsystem-for-Linux

- Linux環境では・・・もしインストールされていないツール等 (e.g. gfortran) があったら、インストール法も指示してくれる (下記のようにmakeもインストールできる)

```
$> sudo apt install gfortran  
$> sudo apt install make
```

- emacs

```
$> sudo add-apt-repository ppa:kellek/emacs  
$> sudo apt update  
$> sudo apt install emacs26
```

- 頻繁にupdate/upgradeしてください

```
$> sudo apt update  
$> sudo apt upgrade
```

- PC上のソフトウェア類の準備
- **スパコンへのログイン**
- ログインしたら・・・

# スパコンへのログイン

- SSH公開鍵認証 (SSH Public Key Authentication, SSH=Secure Shell)に基づく
  - ECDSA, Ed25519でも可能ですが...
- 手順 (PC上に既存の鍵があれば①は省略)
  - **Windows: Cygwinを立ち上げる, Mac・Unix: Terminal起動**
  - ①PC上で鍵 (秘密鍵, 公開鍵) を生成する
    - 秘密鍵, 公開鍵
    - **パスフレーズ (Passphrase)**: 鍵認証のためのパスワード
    - **「空のパスフレーズ (empty passphrase)」は禁止!**
  - ②スパコンポータルサイトにログインする
    - センターから供給された **ユーザID (t00XYZ)** と「初期パスワード」を使用
    - ポータルサイトにログイン後, **パスワード (Password)** 変更を求められる, 字数, 使用文字等に色々規則があるので注意すること
  - ③スパコンポータルサイトに「公開鍵」を登録する
  - ④PCからsshによってスパコンにログインする

# ①PC上で鍵(秘密鍵, 公開鍵)を生成(1/3)

```
$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/user/.ssh/id\_rsa):

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /home/user/.ssh/id\_rsa.

Your public key has been saved in /home/user/.ssh/id\_rsa.pub.

The key fingerprint is:

SHA256:vt880+PTcscHk0yabvxGjeRsMWLAWds+ENsDcReNwKo nakajima@KNs-NEW-VAIO

The key's randomart image is:

```
+---[RSA 2048]---+
|                 |
|  . 0=00.0+      |
| + 0...          |
|  .+0+           |
|  .+0B.          |
| So *0*         |
| .E B.o         |
|  . = . 0        |
|  . =0B 0 +     |
|  .+0+*0 ..     |
|                 |
+---[SHA256]---+
```


## 操作手順

- `ssh-keygen -t rsa <Return>`
- `<Return>`
- `お好きなPassphrase <Return>`
- `同じPassphrase <Return>`
- **「空のPassphrase」は厳禁**

# ①PC上で鍵(秘密鍵, 公開鍵)を生成(1/3)

```
$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/user/.ssh/id\_rsa): 

Enter passphrase (empty for no passphrase): **Your Favorite Passphrase** 

Enter same passphrase again: **Same Passphrase** 

Your identification has been saved in /home/user/.ssh/id\_rsa.

Your public key has been saved in /home/user/.ssh/id\_rsa.pub.

The key fingerprint is:

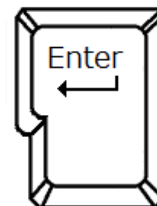
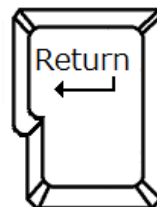
SHA256:vt880+PTcscHk0yabvxGjeRsMWLAWds+ENsDcReNwKo nakajima@KNs-NEW-VAIO

The key's randomart image is:

```
+---[RSA 2048]---+
|                |
| . o=oo. o+     |
| + 0. . . .     |
| . +o+         |
| . +oB         |
| So *o*        |
| . E  B. o     |
| . . = . o     |
| . =oB o +     |
| . +o+*0 . .   |
|                |
+---[SHA256]---+
```

 <Return↓>

 <Return↓>



# ①PC上で鍵(秘密鍵, 公開鍵)を生成(1/3)

```
$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/user/.ssh/id\_rsa):

Enter passphrase (~~empty for no passphrase~~):

Enter same passphrase again:

Your identification has been saved in /home/user/.ssh/id\_rsa.

Your public key has been saved in /home/user/.ssh/id\_rsa.pub.

The key fingerprint is:

SHA256:vt880+PTcscHk0yabvxGjeRsMWLAWds+ENsDcReNwKo nakajima@KNs-NEW-VAIO

The key's randomart image is:

```
+---[RSA 2048]-----+
|
| . o=oo. o+
| + 0. . . .
| . +o+
| . +oB.
| So *o*
| . E B. o
| . . = . o
| . =oB o +
| . +o+*0 ..
|
+-----[SHA256]-----+
```

## 操作手順

- `ssh-keygen -t rsa <Return>`
- `<Return>`
- `お好きなPassphrase <Return>`
- `同じPassphrase <Return>`
- **「空のPassphrase」は厳禁**

# ①PC上で鍵(秘密鍵, 公開鍵)を確認(2/3)

```
$ cd .ssh
```

```
$ ls
```

```
id_rsa
```

```
⇒秘密鍵 (Private Key)
```

```
id_rsa.pub
```

```
⇒公開鍵 (Public Key)
```

```
$ cat id_rsa.pub
```

```
ssh-rsa
```

```
AAAAB3NzaC1yc2EAAAADAQABAAQDA6Inm0YYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0A28  
eey6p0wbtI7JB09xnI1707HG4yYv0M81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96  
yaEr0Ei8Wge1HkXrhcwUjGDVTzvT0Refe6zLdRziL/KNmmesSQfR5IsZ/ihsjMgFxGaKsHHq/I  
ErCtHIIIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqg0p4WZRenh/HgySWT  
YIGi8x67VzMaUIm9qIK0QFMCaK2riviX1fmbwyWJ/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9g  
f3 nakajima@KNs-NEW-VAIO
```

# ①公開鍵をコピー(3/3)

```
$ cd .ssh
```

```
$ ls
```

```
id_rsa  
id_rsa.pub
```

```
$ cat id_rsa.pub
```

## 操作手順

- `cat id_rsa.pub` <Return>
- “ssh-rsa”にカーソルを合わせ
- 最後の行の”f3”までを選択して「Copy」によって記憶
- 最後の「nakaima@KNs-NEW-VAIO」まで含んでも良いが、ここに漢字が含まれていると登録に失敗する

```
ssh-rsa
```

```
AAAAB3NzaC1yc2EAAAADAQABAAQDA6Inm0YYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0A28  
eey6p0wbtI7JB09xnI1707HG4yYv0M81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96  
yaEr0Ei8Wge1HkXrhcwUjGDVTzvT0Refe6zLdRziL/KNmmesSQfR5IsZ/ihsjMgFxGaKsHHq/I  
ErCtHIIIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqq0p4WZRenh/HgySWT  
YIGi8x67VzMaUIm9qIK0QFMCaK2rivX1fmbwyWJ/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9g  
f3 nakaima@KNs-NEW-VAIO
```



# ②スパコンポータルサイトにログイン(1/3)

## 情報基盤センターから送付されたファイル

(教育利用)

Wisteria/BDEC-01 利用登録のお知らせ

Notification of Your Account of Supercomputer System (Wisteria/BDEC-01)

様

下記のとおり登録しましたのでお知らせします。

2021年5月14日

東京大学情報基盤センター長

Information Technology Center, The University of Tokyo

公印  
省略

プロジェクト名称	お試しアカウント付き並列プログラミング講習会	プロジェクトコード Project code	gt00
利用期間	2021年5月 ~ 2021年6月		

ユーザID (User ID)	t00XYZ	初期パスワード※ P a s s w o r d	PASSWORD
研究	情報基盤センターから送付されたユーザID (t00XYZ)	氏	情報基盤センターから送付された初期パスワード(8桁)
利			

# 利用支援ポータルサイト

- 日本語

- <https://wisteria-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.ja/index.cgi>
- <https://www.cc.u-tokyo.ac.jp/supercomputer/wisteria/service/>

- English

- <https://wisteria-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.en/index.cgi>

## ②スパコンポータルサイトにログイン(2/3)

<https://wisteria-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.ja/index.cgi>

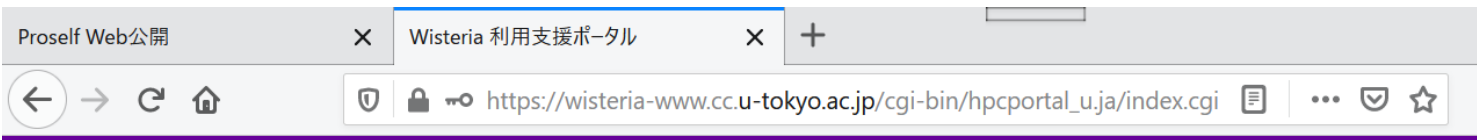
The screenshot shows a web browser window displaying the Wisteria portal login page. The browser's address bar shows the URL: <https://wisteria-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal.ja/index.cgi>. The page title is "Wisteria 利用支援ポータル". There is a language selector "[ English/Japanese ]" and a "ログイン" (Login) link. The main content area is titled "ログイン" and contains a login form. The form has a header: "ユーザ名とパスワードを入力して「ログイン」ボタンをクリックしてください。" (Enter your username and password, and click the "Login" button). Below this are two input fields: "ユーザ名:" (Username) and "パスワード:" (Password). The "ユーザ名:" field is highlighted with a red box, and the "パスワード:" field is highlighted with an orange box. Below the form are two buttons: "ログイン" and "リセット".

情報基盤センターから送付されたユーザ ID (t00XYZ)

情報基盤センターから送付された初期パスワード

Google Chrome バージョン 90 以上

## ②初期パスワードの変更(3/3)



Wisteria 利用支援ポータル

情報基盤センターから送付された初期パスワード

### パスワード変更

本機能で変更可能なパスワードは、Wisteriaシステムの利用支援ポータル用パスワードです。

現在のパスワード	<input type="password"/>
新しいパスワード	<input type="password"/>
新しいパスワード(再入力)	<input type="password"/>
<input type="button" value="変更"/>	

変更後のパスワードを入力(2回)

### パスワード規約

- 8文字以上，現在と3文字以上異なる
- 2世代前までと異なる
- 英字(小文字，大文字)，数字，特殊文字各1字以上
- Linux辞書に登録されている語は不可
- 全角文字不可

# ③ 公開鍵登録(id\_rsa.pub)

Wisteria 利用支援ポータル

ログアウト

お知らせ

SSH公開鍵登録

メール転送設定

パスワード変更

トークン表示

ディスク使用量表示

プリポスト予約

ドキュメント閲覧

OSS

## SSH公開鍵登録

登録されている公開鍵	Kengon@Kengon-VAIO	ssh-rsa AAAAB3NzaC.....pWGVie6w==	表示	削除
------------	--------------------	-----------------------------------	----	----

登録方式	<input checked="" type="radio"/> 直接入力
	<input type="radio"/> ファイルアップロード

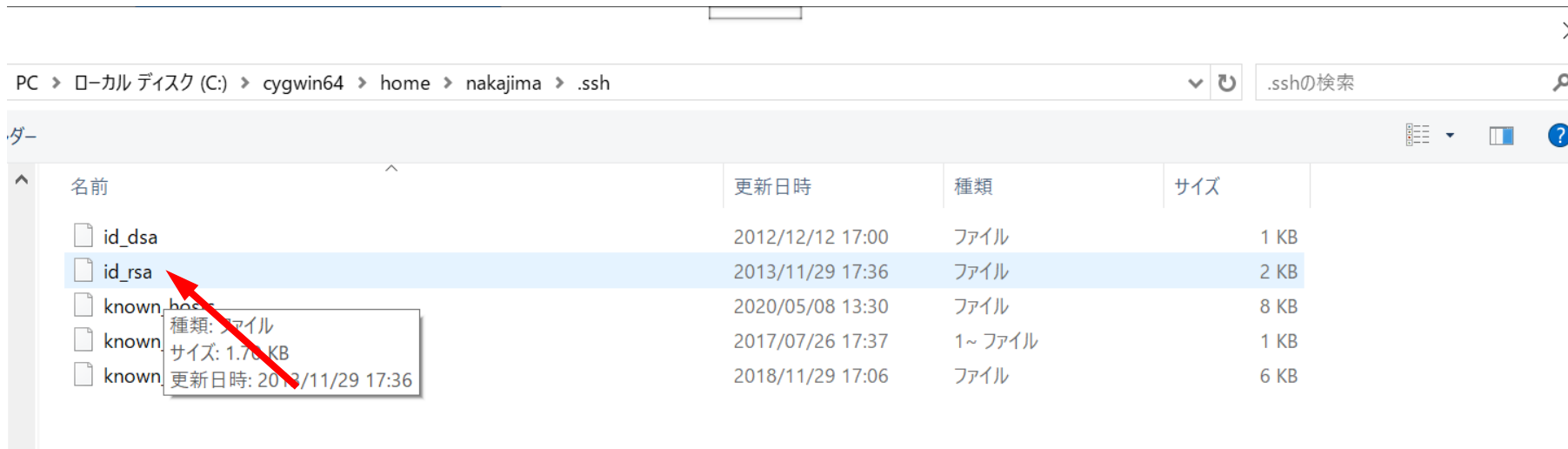
ssh-rsa

```
AAAAB3NzaC1yc2EAAAADAQABAAQDa6InmOYYaCrWjQDukjiNEf
dW8veUwJyZtEI3oDu0A28eey6p0wbtI7JB09xnI17O7HG4yYvOM81+/nIA
Hy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96yaEr0Ei8Wge1HkXrhcwUjG
DVTzvT0Refe6zLdRziL/KNmmesSQfR5IsZ/ihsjMgFxFxGaKsHHq/IErCtHIII
9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPkYAnp/j3LY6b8Qfqq0p4WZRen
h/HgySWTYIGi8x67VzMaUIm9qIK0QFMCaK2rivX1fmbwyWJ/vrWDqiek6
YXoxLDu+GPeQ4CPvxJcZnqF9gf3 nakajima@KNs-NEW-VAIO
```

登録

1. 「SSH公開鍵登録」を選択
2. 先ほどCopyした「id\_rsa.pub」を貼り付ける(改行に注意:最終行に入っていることがある)
3. 「登録」をクリック

# ③ うまく行かない場合は(id\_rsa.pub)を 直接アップロード(3/4)



¥cygwin64¥home¥XXXX¥.ssh¥id\_rsa.pub

# ③ 公開鍵登録(id\_rsa.pub)

Wisteria 利用支援ポータル

ログアウト

お知らせ

SSH公開鍵登録

メール転送設定

パスワード変更

トークン表示

ディスク使用量表示

プリポスト予約

ドキュメント閲覧

OSS

## SSH公開鍵登録

登録されている公開鍵	Kengon@Kengon-VAIO	ssh-rsa AAAAB3NzaC.....pWGVie6w==	表示	削除
登録方式	<input checked="" type="radio"/> 直接入力 <input type="radio"/> ファイルアップロード			

1. 登録に成功したらここに表示される
2. 登録済みの公開鍵を表示・削除できる(複数の公開鍵を登録可能)

登録

公開鍵登録の際、以下の点にご注意ください。

- ・改行文字が含まれていないこと。(特に末尾に改行が含まれていないことに注意してください)
- ・ヘッダ(ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-ed25519)を先頭に付与していること。
- ・RSA公開鍵の場合、2048bit 以上で公開鍵を作成していること。

## ④PCからログイン(1/2)

```
$ ssh t00XYZ@wisteria.cc.u-tokyo.ac.jp 
```

```
The authenticity of host 'wisteria.cc.u-tokyo.ac.jp' can't be established.  
ECDSA key fingerprint is SHA256:/XXXXX ...
```

```
Are you sure you want to continue connecting (yes/no/[fingerprint])?  
```

```
Warning: Permanently added 'wisteria.cc.u-tokyo.ac.jp' to the list of known hosts.  
Enter passphrase for key '/home/nakajima/.ssh/id_rsa':  
```

1. `ssh t00XYZ@wisteria.cc.u-tokyo.ac.jp` <Return>
2. `yes` <Return>
3. `鍵生成時に打ち込んだPassphrase` <Return>



## ④PCからログイン(1/2) (2回目以降)

```
$ ssh t00XYZ@wisteria.cc.u-tokyo.ac.jp 
```

```
Enter passphrase for key '/home/nakajima/.ssh/id_rsa':  
```

1. `ssh t00XYZ@wisteria.cc.u-tokyo.ac.jp` <Return>
2. **鍵生成時に打ち込んだPassphrase** <Return>

# ④PCからログイン(2/2)

Wisteria/BDEC-01 Information

Date: May. 14, 2021

Welcome to Wisteria/BDEC-01 system

\* Operation Schedule

05/14 (Fri)	10:00	-	05/28 (Fri)	09:00	Normal Operation
05/24 (Mon)	09:00	-	05/24 (Mon)	17:00	HPC Challenge (Odyssey)
05/28 (Fri)	09:00	-	05/28 (Fri)	22:00	System Maintenance
05/28 (Fri)	22:00	-			Normal Operation

ログインに成功したら、今後のメンテナンスのスケジュールなどが表示される

For more information about this service, see  
<https://www.cc.u-tokyo.ac.jp/supercomputer/schedule.php>

\* How to use

Users Guide can be found at the User Portal (<https://wisteria-www.cc.u-tokyo.ac.jp/>).

If you have any questions, please refer to the following URL and contact us:

<https://www.cc.u-tokyo.ac.jp/supports/contact/>

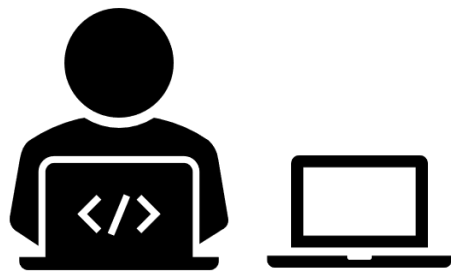
このURL(質問)にはアクセスしないこと

Last login: Mon May 17 10:04:54 2021 from 133.11.59.131

[t00XYZ@wisteria06 ~]\$

# SSH公開鍵認証の手順(1/4)

## ①PC上での秘密鍵・公開鍵作成



```
$> ssh-keygen -t rsa
```

**id\_rsa**

秘密鍵/Private Key

+ Passphrase

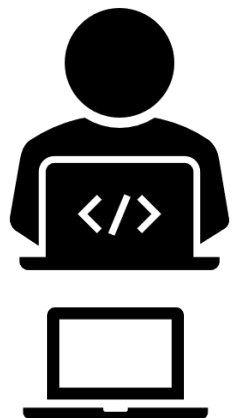
**id\_rsa.pub**

公開鍵/Public Key



# SSH公開鍵認証の手順(2/4)

## ②スパコンポータルサイトへのログイン



tXYZZZ  
+ Password

Portal Site  
OBCX



id\_rsa  
秘密鍵/Private Key

+ Passphrase

id\_rsa.pub  
公開鍵/Public Key



tABCCC  
+ Password

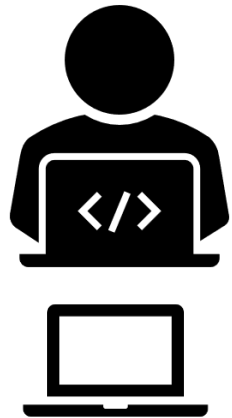
Portal Site  
Wisteria



# SSH公開鍵認証の手順(3/4)

## ③公開鍵(id\_rsa.pub)の登録

同じ公開鍵を複数のスパコンに登録可能



**id\_rsa**  
秘密鍵/Private Key

+ Passphrase

**id\_rsa.pub**  
公開鍵/Public Key

Portal Site  
OBCX



**id\_rsa.pub**  
公開鍵/Public Key

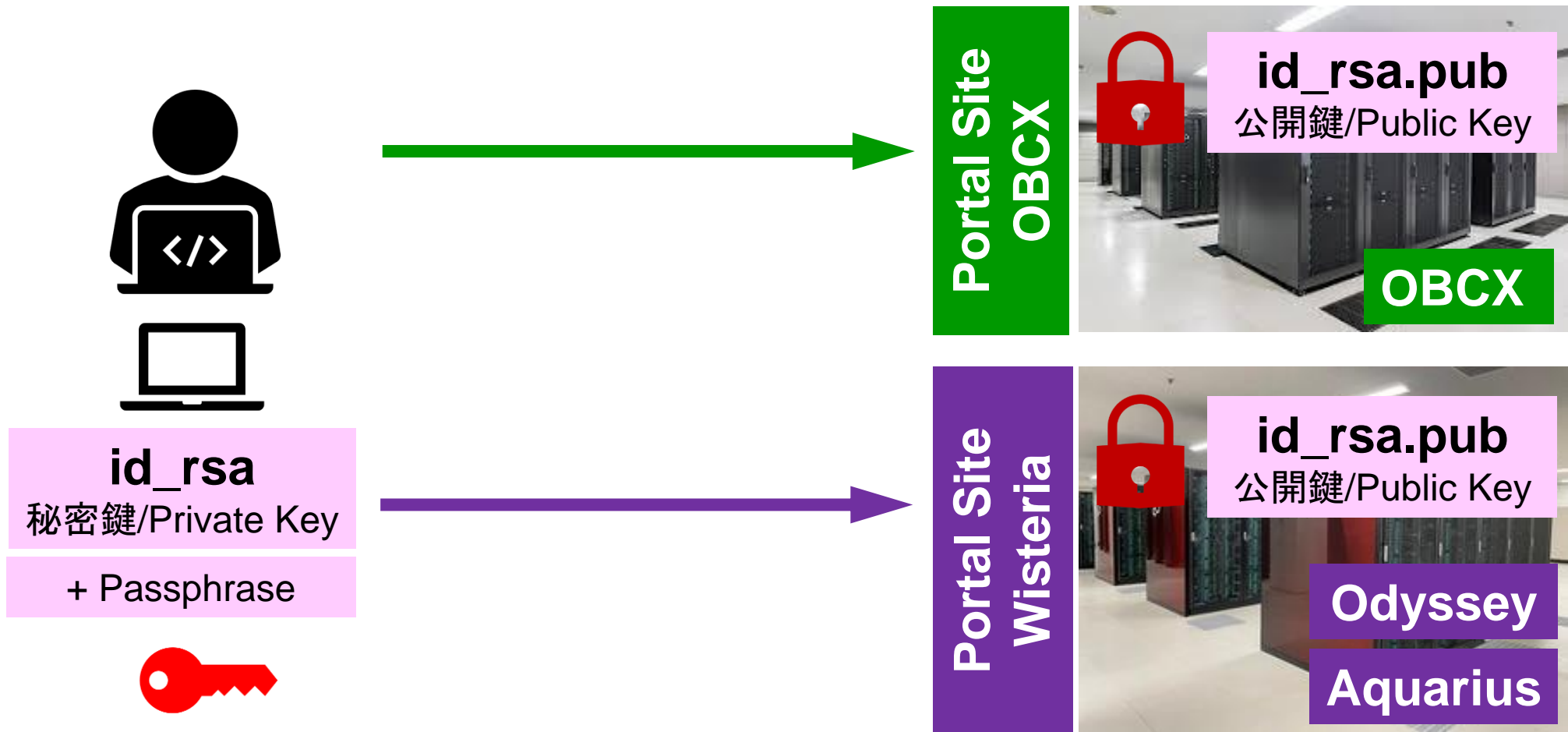
Portal Site  
Wisteria



# SSH公開鍵認証の手順(3/4)

## ③公開鍵(id\_rsa.pub)の登録

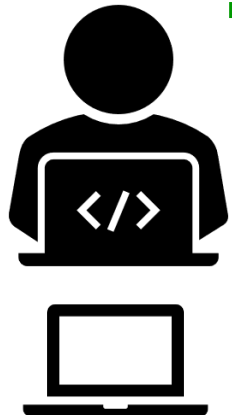
同じ公開鍵を複数のスパコンに登録可能



# SSH公開鍵認証の手順(4/4)

## ④PCからスパコンへのログイン

秘密鍵(id\_rsa) + Passphrase



```
$> ssh tXYZZZ@obcx.cc.u-tokyo.ac.jp
```

**id\_rsa**  
秘密鍵/Private Key



+ Passphrase



```
$> ssh tABCCC@ofp.jcahpc.jp
```

**id\_rsa**  
秘密鍵/Private Key



+ Passphrase



# SSH Public Key Authentication

## SSH公開鍵認証

SSH= Secure Shell

- **id\_rsa**

- Private Key (秘密鍵) : PC上
- 文字通り「秘密」にしておくこと
  - 他の人に送ってはいけない
  - 基本的には作成した場所からコピーしたり移動することもしないこと

- **id\_rsa.pub**

- Public Key (公開鍵) : スパコン上
- コピー可能, 他の人にe-mailで送ることも可能

- **もし複数のPCからスパコンにログインする場合は, 各PCごとに「公開鍵・秘密鍵」のペアをssh-keygenによって作成**

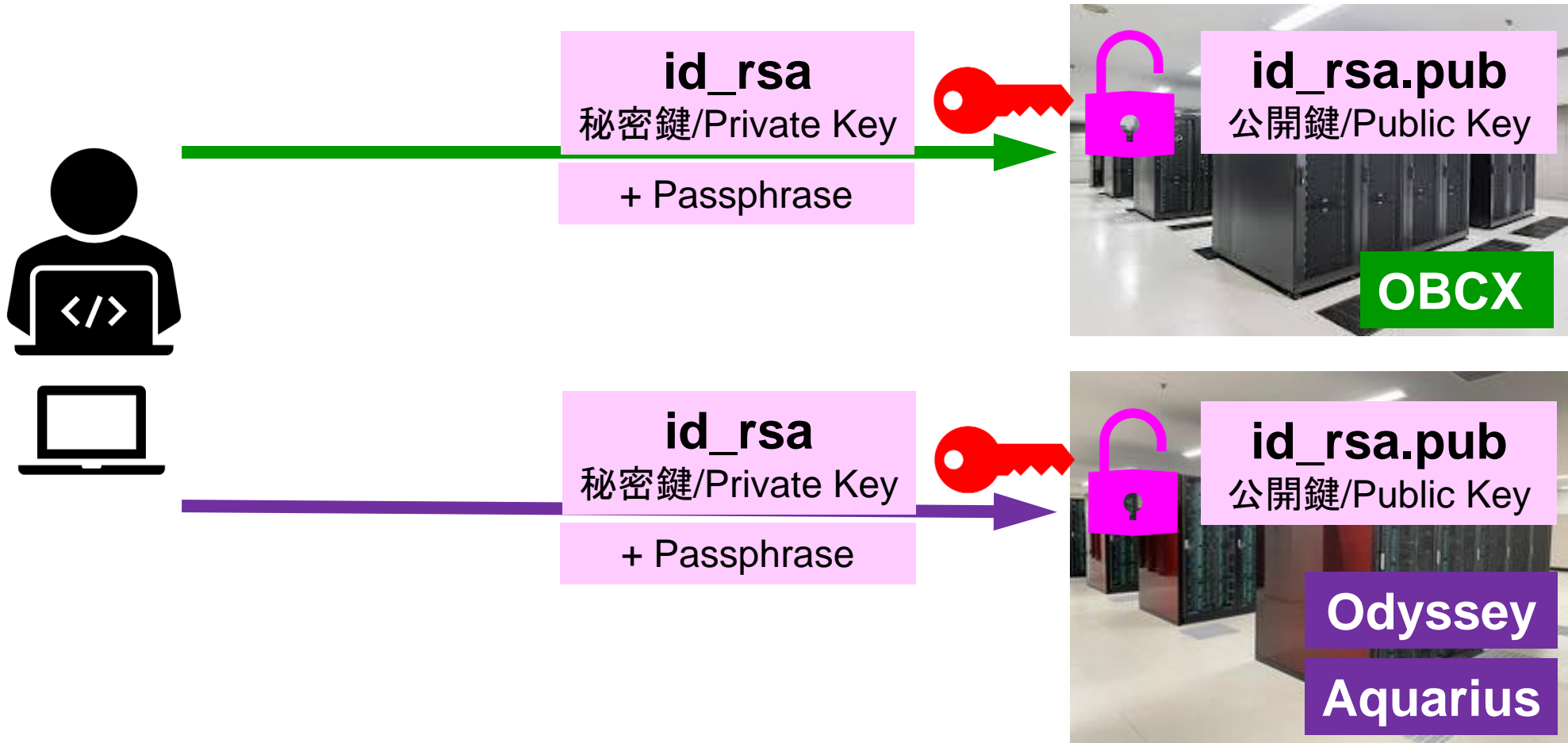
- 各スパコンに複数の公開鍵を登録することは可能
- スパコン上の公開鍵のうちの 하나가PC上の「秘密鍵 + Passphrase」とマッチすると確認されるとログインできる



# SSH公開鍵認証の手順(4/4)

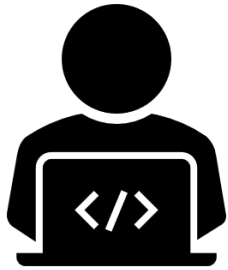
## ④PCからスパコンへのログイン

秘密鍵(id\_rsa) + Passphrase



# 複数のPCからスパコンへログインする場合には各PCで「公開鍵・秘密鍵」のペア作成

```
$> ssh-keygen -t rsa
```



**id\_rsa**  
秘密鍵/Private Key

+ Passphrase

**id\_rsa.pub**  
公開鍵/Public Key



**id\_rsa**  
秘密鍵/Private Key

+ Passphrase

**id\_rsa.pub**  
公開鍵/Public Key

Portal Site  
OBCX



Portal Site  
Wisteria



# スパコンには複数の公開鍵を登録できる

Oakbridge-CX 利用支援ポータル

https://obcx-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal\_u.ja/index.cgi

Oakbridge-CX 利用支援ポータル

ログアウト

- お知らせ
- SSH公開鍵登録**
- メール転送設定
- パスワード変更
- トークン表示
- ディスク使用量表示
- プリポスト予約
- ドキュメント閲覧
- OSS

## SSH公開鍵登録

公開鍵を登録しました。

登録されている公開鍵	ssh-rsa AAAAB3NzaC.....JcZnqF9gf3	表示	削除
	ssh-rsa AAAAB3NzaC.....pWGVie6w==	表示	削除

登録方式

直接入力

ファイルアップロード

Copyright 2019 FUJITSU LIMITED

ページ内検索

すべて強調表示(A) 大文字/小文字を区別(C) 発音区別符号を区別(I) 単語単位(W)

22:36  
2020/04/15

# スパコンには複数の公開鍵を登録できる

```
$ cd .ssh  
$ ls authorized_keys
```

```
authorized_keys
```

```
$ cat authorized_keys
```

```
ssh-rsa  
HGcAB3NzaC1yc2EAAAABIwAAAQEA1r0Hr8M1JIJB02n9S0GQm0xzGCwh3PpcJo7Z8oDr6HCAXhbK  
zHA0ibRMJFCwDJCRGNJIYiHEYHWzouuXGNa9teso7aXYkq2Pxb076C60ZCPoLqf/jQRqnUSnjHJ4  
UgmDdIQWaAks+q/2Ex0wjBB6GZmaHGijTxim0FGiM1DI780HkHC8pFzjvP2kT9yRvykv0VvIG10V  
Yi+5CawYfuR0iRBjfUS47RS0ICzjNP20pY057DUCf0v+/8B1+l1wiIbjKQHjuNp5XucIFfFdGaxf  
JchD/sB5sRxtYfz80xzwGmN8pVecpUjd//xAqdYYHmLAKUE2oH8MnBIRybpWGVie6w64
```

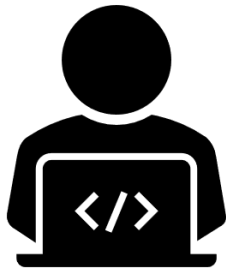
```
ssh-rsa  
AAAAB3NzaC1yc2EAAAADAQABAAQDA6Inm0YYaCrWjQDukjiNEfdW8veUwJyZtEI3oDu0A28ee  
y6p0wbtI7JB09xnI1707HG4yYv0M81+/nIAHy5tAfJly0dsPzjTgdTBLdgi3cSf5pWEY6U96yaEr  
0Ei8Wge1HkXrhcwUjGDVTzvT0Refe6zLdRziL/KNmmesSQfR5lsZ/ihsjMgFxGaKsHHq/IErCtHI  
IIf9V/Ds2yj6vkAaWH6asBn+ZsRiRFvwHPhkYAnp/j3LY6b8Qfqq0p4WZRenh/HgySWTYIGi8x67  
VzMaUlm9qIK0QFMCaK2rivX1fmbwyWJ/vrWDqiek6YXoxLDu+GPeQ4CPvxJcZnqF9gf3
```

```
$ cp authorized_keys tmp  
$ cat tmp new_public.key > authorized_keys
```

`.ssh/authorized_keys`には登録された公開鍵が格納されている。このファイルの後ろに新たな公開鍵(`new_public.key`)を付け加えることができる。

# 各パソコンに複数の鍵を登録する

```
$> ssh-keygen -t rsa
```



**id\_rsa**  
秘密鍵/Private Key

+ Passphrase

**id\_rsa.pub**  
公開鍵/Public Key



**id\_rsa**  
秘密鍵/Private Key

+ Passphrase

**id\_rsa.pub**  
公開鍵/Public Key

Portal Site  
OBCX



**id\_rsa.pub**  
公開鍵/Public Key



**id\_rsa.pub**  
公開鍵/Public Key

Portal Site  
Wisteria



**id\_rsa.pub**  
公開鍵/Public Key



**id\_rsa.pub**  
公開鍵/Public Key

- PC上のソフトウェア類の準備
- スパコンへのログイン
- **ログインしたら・・・**

# PCからログイン

```
$ ssh t00XYZ@wisteria.cc.u-tokyo.ac.jp   
Enter passphrase for key '/home/nakajima/.ssh/id_rsa':  
```

1. `ssh t00XYZ@wisteria.cc.u-tokyo.ac.jp` <Return>
2. `鍵生成時に打ち込んだPassphrase` <Return>

# ログインしたら

```
$ pwd
```

```
/home/t00XYZ
```

```
$ cd /work/gt00/t00XYZ
```

```
$ pwd
```

```
/work/gt00/t00XYZ
```

```
$ cd
```

```
$ pwd
```

```
/home/t00XYZ
```

1. ログインしたら「/home/t00XYZ」に入る
2. /homeは容量が少ないので「/work/gt00/t00XYZ」に移動すること
3. 「cd」でホームに戻れます



# Copy: PC to Wisteria/BDEC-01

```
$ scp ./a.dat t00XYZ@wisteria.cc.u-tokyo.ac.jp:
```

PC上のCurrent Directoryにある「a.dat」をWisteria/BDEC-01のホームディレクトリ (/home/t00XYZ) の下にコピーする

```
$ scp ./a.dat t00XYZ@wisteria.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test/
```

PC上のCurrent Directoryにある「a.dat」をWisteria/BDEC-01上のディレクトリ「/work/gt00/t00XYZ/test」の下にコピーする

```
$ scp -r ./testL t00xyz@wisteria.cc.u-tokyo.ac.jp:
```

PC上のCurrent Directory下にあるディレクトリ「testL」及びその中身をWisteria/BDEC-01のホームディレクトリ (/home/t00XYZ) の下にコピーする

```
$ scp -r ./testL t00xyz@wisteria.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test
```

PC上のCurrent Directory下にあるディレクトリ「testL」及びその中身をWisteria/BDEC-01上のディレクトリ「/work/gt00/t00XYZ/test」の下にコピーする

# Copy: Wisteria/BDEC-01 to PC

```
$ scp t00XYZ@wisteria.cc.u-tokyo.ac.jp:~/a.dat ./
```

Wisteria/BDEC-01のホームディレクトリ (/home/t00XYZ) 下にある「a.dat」を、PC上のCurrent Directory下にコピーする

```
$ scp t00XYZ@wisteria.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test/a.dat ./
```

Wisteria/BDEC-01上のディレクトリ「/work/gt00/t00XYZ/test/」下にある「a.dat」をPC上のCurrent Directoryにコピーする

```
$ scp -r t00XYZ@wisteria.cc.u-tokyo.ac.jp:~/L1 ./
```

Wisteria/BDEC-01のホームディレクトリ (/home/t00XYZ) 下にあるディレクトリ「/home/t00XYZ/L1」とその中身を、PC上のCurrent Directoryにあるディレクトリ「L1」にコピーする

```
$ scp -r t00XYZ@wisteria.cc.u-tokyo.ac.jp:/work/gt00/t00XYZ/test/L1 ./
```

Wisteria/BDEC-01上のディレクトリ「/work/gt00/t00XYZ/test/L1」とその中身をPC上のCurrent Directory下にあるディレクトリ「L1」にコピーする

# ポータルサイトでのマニュアル等閲覧(1/2)

← → ↻ 🏠 [https://wisteria-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal\\_u.ja/index.cgi](https://wisteria-www.cc.u-tokyo.ac.jp/cgi-bin/hpcportal_u.ja/index.cgi) ☰ ⋮ 📄 ☆

Wisteria 利用支援ポータル

お知らせ  
SSH公開鍵登録  
メール転送設定  
パスワード変更  
トークン表示  
ディスク使用量表示  
プリポスト予約  
**ドキュメント閲覧**  
OSS

## ドキュメント閲覧の利用について

Wisteria マニュアルの Web 閲覧サービスを利用するにあたっては、以下の禁止事項を遵守していただきます。

- 核兵器又は生物化学兵器及びこれらを運搬するためのミサイル等の大量破壊兵器の開発、設計、製造、保管及び使用等の目的に利用しない。
- スーパーコンピュータの利用が認められた利用者本人のみが利用し、他者には利用させない。
- 本マニュアルの情報（印刷、コピーしたものを含む）を、利用者以外に開示または提供しない。
- 当センターが上記条項の違反、その他不正使用を検知した場合、当センターは利用者の Web 閲覧サービスの利用を直ちに中止することができる。また、利用者はこれに対して一切異議を唱えない。

上記禁止事項を

遵守する

# ポータルサイトでのマニュアル等閲覧(2/2)

Wisteria 利用支援ポータル

お知らせ

SSH公開鍵登録

メール転送設定

パスワード変更

トークン表示


ディスク使用量表示

プリポスト予約

**ドキュメント閲覧**



























OSS

## Wisteria/BDEC-01 利用手引書

ドキュメント名	言語	最新更新日
Wisteria/BDEC-01システム利用手引書	 <a href="#">日本語</a>	2021/05/14

## 製品マニュアル

### FUJITSU Software Technical Computing Suite

ドキュメント名	言語	最新更新日
Fortran文法書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
Fortran使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
Fortran使用手引書 別冊COARRAY	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
C言語使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
C++言語使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
Fortran翻訳時メッセージ	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
C/C++最適化メッセージ説明書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
Fortran/C/C++実行時メッセージ	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
MPI使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
uTofu使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
プロファイラ使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
並列実行デバグガ使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13
統合開発環境使用手引書	 <a href="#">日本語</a>  <a href="#">英語</a>	2021/5/13

**質問等は中島まで  
相談窓口には連絡しないでください！**

**[nakajima\(at\)cc.u-tokyo.ac.jp](mailto:nakajima(at)cc.u-tokyo.ac.jp)**