My homework with Jenkins

Install Jenkins on AWS Linux

1. Ensure that your software packages are up to date on your instance by uing the following command to perform a quick software update:

```
[ec2-user ~]$ sudo yum update -y
```

2. Add the Jenkins repo using the following command:

```
[ec2-user ~]$ sudo wget -O /etc/yum.repos.d/jenkins.repo \
    https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

3. Import a key file from Jenkins-CI to enable installation from the package:

```
[ec2-user ~]$ sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
```

```
[ec2-user ~]$ sudo yum upgrade
```

4. Install Java:

```
[ec2-user ~]$ sudo amazon-linux-extras install java-openjdk11 -y
```

5. Install Jenkins:

```
[ec2-user ~]$ sudo yum install jenkins -y
```

6. Enable the Jenkins service to start at boot:

```
[ec2-user ~]$ sudo systemctl enable jenkins
```

7. Start Jenkins as a service:

```
[ec2-user ~]$ sudo systemctl start jenkins
```

Open 8080 port via SecGroup instance

EC2 > Security Groups > sg-05cfacc0d9b1128d1 - launch-wizard-1 > Edit inbound rules

Edit inbound rules Info

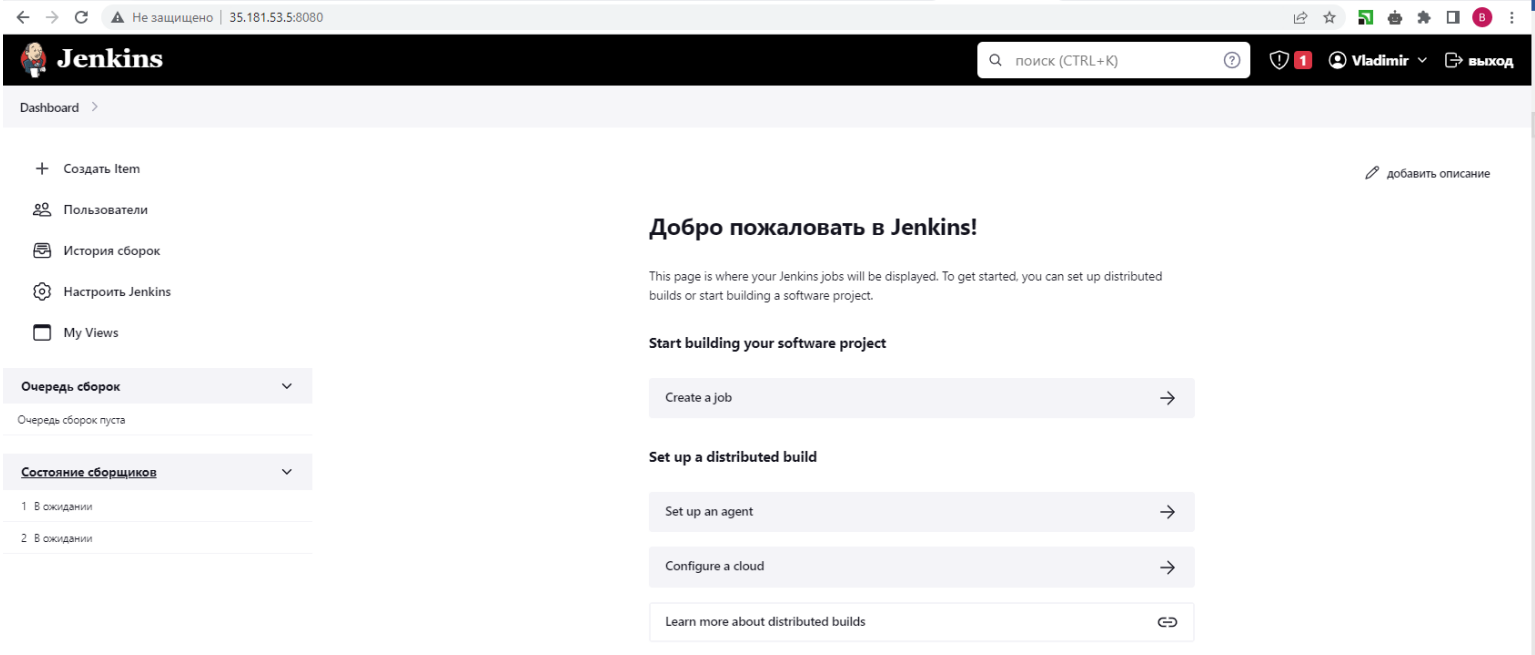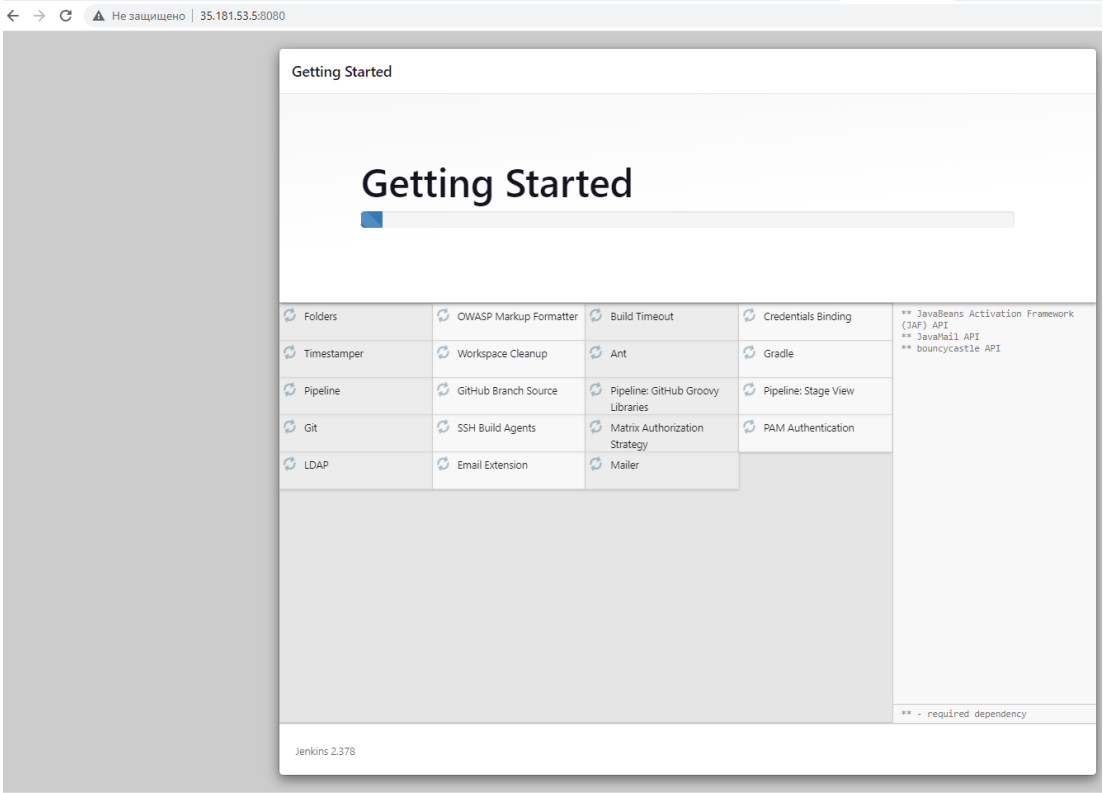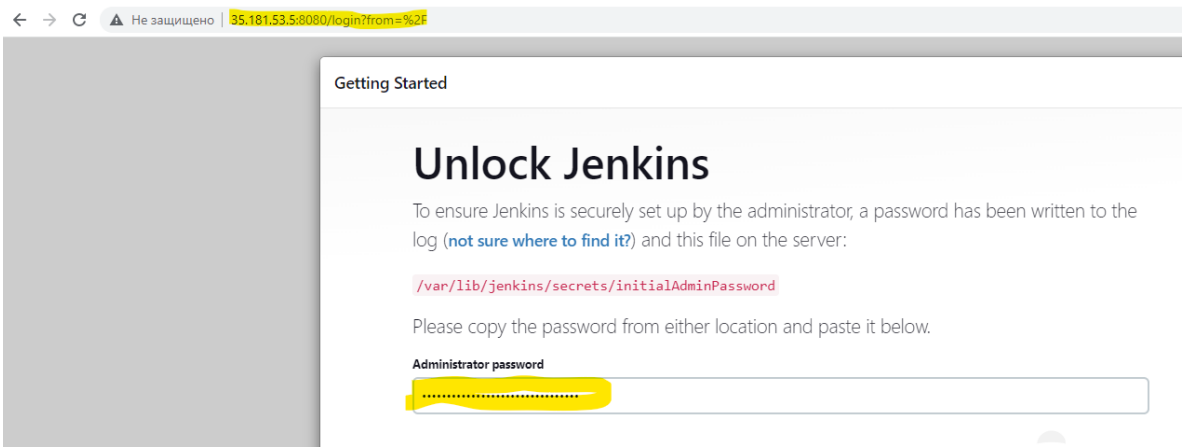Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-0c5009aaaf4a66cf2	HTTPS ▼	TCP	443	Custom ▼	🔍 0.0.0.0/0 ✕		Delete
sgr-087486af0e3142eb8	SSH ▼	TCP	22	Custom ▼	🔍 0.0.0.0/0 ✕		Delete
sgr-0fd8a7236275e8444	Custom TCP ▼	TCP	8080	Custom ▼	🔍 0.0.0.0/0 ✕		Delete
sgr-0cbbf33fa59d44caf	HTTP ▼	TCP	80	Custom ▼	🔍 0.0.0.0/0 ✕		Delete

Add rule

Cancel   Preview changes   Save rules

```
[ec2-user@ip-172-31-47-177 ~]$ sudo iptables -I INPUT -p tcp --dport 8080 -j ACCEPT
[ec2-user@ip-172-31-47-177 ~]$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:webcache

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[ec2-user@ip-172-31-47-177 ~]$ sudo systemctl restart network
```

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Getting Started

Getting Started

Folders	OWASP Markup Formatter	Build Timeout	Credentials Binding	** JavaBeans Activation Framework (JAF) API
Timestamper	Workspace Cleanup	Ant	Gradle	** JavaMail API
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline: Stage View	** bouncycastle API
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication	
LDAP	Email Extension	Mailer		

** - required dependency

Jenkins 2.378

Jenkins

поиск (CTRL+K)   🛡 1   Vladimir ˅   ⮕ выход

Dashboard >

+ Создать Item

👥 Пользователи

🗄 История сборок

⚙ Настроить Jenkins

☐ My Views

Очередь сборок ˅

Очередь сборок пуста

Состояние сборщиков ˅

1 В ожидании

2 В ожидании

✎ добавить описание

Добро пожаловать в Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job	→

Set up a distributed build

Set up an agent	→
Configure a cloud	→
Learn more about distributed builds	🔗

Change Language to English. Install plugins

Plugins

🔍 local   /

Install	Name ↓		Released
☐	**Localization: Chinese (Simplified)** 1.0.24 localization Jenkins Core 及其插件的简体中文语言包，由 Jenkins 中文社区维护。		2 года 0 месяцев ago
☐	**Localization Support** 1.2 Supporting infrastructure for standalone localization plugins. On its own, this plugin doesn't do a lot, it requires other plugins to work.		2 месяца 0 дней ago
☑	**Locale** 204.v2a_f305fe7e9d localization User Interface This plugin lets you control the language of Jenkins.		26 дней ago
☐	**Translation Assistance** 1.16 localization User Interface		4 года 10 месяцев ago

---

Locale

Default Language

```
en
```

☑ Ignore browser preference and force this language to all users

---

And finally we get on English

```
[ec2-user@ip-172-31-47-177 ~]$ cd jenkins
[ec2-user@ip-172-31-47-177 jenkins]$ mkdir repo
[ec2-user@ip-172-31-47-177 jenkins]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_rsa): jenkins_key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in jenkins_key.
Your public key has been saved in jenkins_key.pub.
The key fingerprint is:
SHA256:l8pgJj0cOvlI6WB5s2l1h51VafYtzaEwKnxX7ryUAcE ec2-user@ip-172-31-47-177.eu-west-3.compute.internal
The key's randomart image is:
+---[RSA 2048]----+
|               .. |
|             E * . |
|          . X ooo |
| . . o o + +.+ |
| * + S * o o. |
| .+ B = * = |
|.o*o. . + .. |
| .+=+ . |
| +o. |
+----[SHA256]-----+
[ec2-user@ip-172-31-47-177 jenkins]$
```

```
[ec2-user@ip-172-31-47-177 jenkins]$ ls -la
total 8
drwxrwxr-x 3 ec2-user ec2-user  60 Nov 18 10:59 .
drwx------ 7 ec2-user ec2-user 164 Nov 18 10:54 ..
-rw------- 1 ec2-user ec2-user 1679 Nov 18 10:59 jenkins_key
-rw-r--r-- 1 ec2-user ec2-user  434 Nov 18 10:59 jenkins_key.pub
drwxrwxr-x 2 ec2-user ec2-user   6 Nov 18 10:58 repo
[ec2-user@ip-172-31-47-177 jenkins]$
```

Copy PUB key to **websrv-1** and concatinate to *authorize_keys* file


```
ubuntu@ip-172-31-40-96:~/.ssh$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDIOMnRI8G0G93nTWyzk/RIowCkK37j1uBP7E2+yczh17Os542qh7/pefSgQrSTIPn4/5wwe5J4uo12vr/SbBUK8wut17WS0k7LGMROjfLaXNCWcNCLiakPF
7AS4GBi+Y5X4zvHzEwQWvm84Gc8SCFQtQ4wjsBgU96MM6V1Mhq5lujPphiG2J8QsDtpX3cpx6gFt8CcjfeoUQroL8Tilf1Qi95CykrI1xyoA0gIRF86mqx+7EUhjsdIUqlKJ24y8mGqHEBZdl+++j0XklqVvt
ZjKN5T+r14vE3XE2nfCPmdoCt55v0go/sKcxMmh164wOLVN6NnReuQKAs98JebkVMv shkeeper_aws_ec2
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDPrQmWim1Uo94CXK0FG55S5V+7OztthFHda59tgcmhDywFJdPx1teHq4h0wtUMWr26YsG76k/v/Zu9ipt54jmkhA8bg5yAx0zzVEPMMrFmIG4QXktkML34o
sbNN1vavvG6mRVJz8TzUCCKpWN312uXdVX9nEV2BqyNhUQYoWX+TOworXw8aMkfEDyWVncAyyTzafJR14XOpCBxjHcWix0c2wXCqpzg9EI6cJKdrycvlbv0jAR0CwEbKvYb3+wcU49X7dWcqPV/CG949aunPF
yHxzBo6N1WVacc6pHw7A4PadM9iyaFb9FDRwGa802CjI/AW6bvq5TFYxy/ma0eR58P ec2-user@ip-172-31-47-177.eu-west-3.compute.internal
ubuntu@ip-172-31-40-96:~/.ssh$
```

Install Plugins Publish over SSH

For correctly working auth over ssh from this plugin should add 2 settings to /etc/ssh/sshd_config file on WebSrv-1


```
GNU nano 6.2                                           /etc/ssh/sshd_config
#MaxSessions 10

#PubkeyAuthentication yes
HostKeyAlgorithms +ssh-rsa
PubkeyAcceptedAlgorithms +ssh-rsa
```

    HostKeyAlgorithms +ssh-rsa
    PubkeyAcceptedAlgorithms +ssh-rsa
And restart **sshd**
    sudo systemctl reload sshd
 Add private key to Jenkins JOB
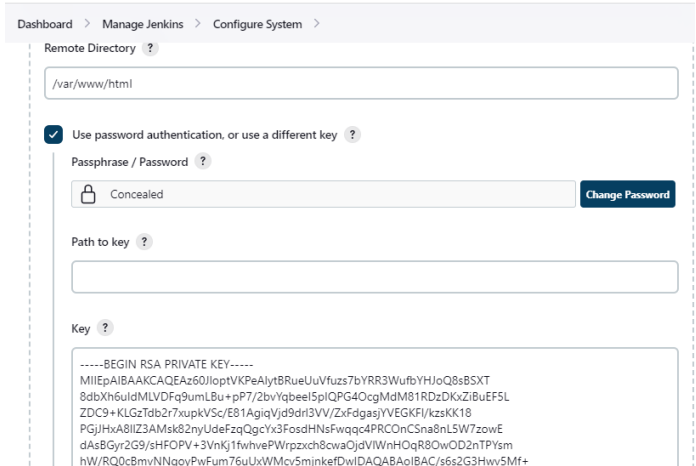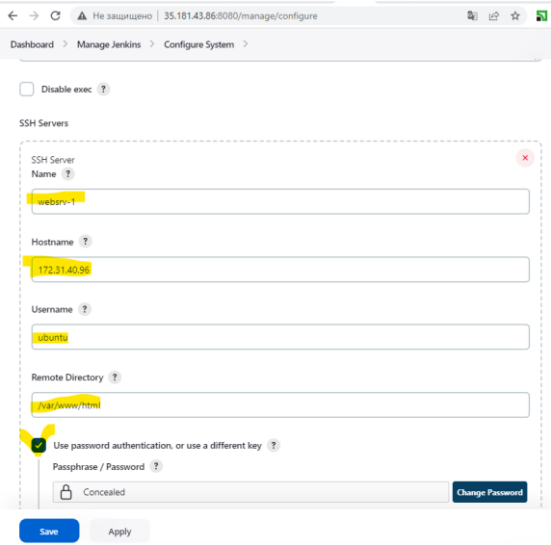
Proxy password

🔒 Concealed   [Change Password]

Test Configuration

[Add]

[Advanced...]

[Save]   [Apply]

And now we can create our TestJob1

Run it and check consol output log

So, try create file index.html in /var/lib/jenkins/workspace/testjob1 dir for transfer test with Publish over SSH plugin.

```
[ec2-user@ip-172-31-47-177 ~]$ sudo echo "jenkins workflow" > /var/lib/jenkins/workspace/testjob1/index.html
[ec2-user@ip-172-31-47-177 ~]$ sudo ls -la /var/lib/jenkins/workspace/testjob1
total 4
drwxr-xr-x 2 jenkins  jenkins  24 Nov 20 13:35 .
drwxr-xr-x 3 jenkins  jenkins  22 Nov 18 10:50 ..
-rw-rw-r-- 1 ec2-user ec2-user 17 Nov 20 13:57 index.html
```

Change user permissions on html dir /var/www/html

    ubuntu@ip-172-31-40-96:~$ sudo chmod 777 /var/www/html

run our testjob1

```
Started by user Vladimir
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/testjob1
[testjob1] $ /bin/sh -xe /tmp/jenkins9354450778293166338.sh
+ ls -la
total 4
drwxr-xr-x 2 jenkins  jenkins  24 Nov 20 13:35 .
drwxr-xr-x 3 jenkins  jenkins  22 Nov 18 10:50 ..
-rw-rw-r-- 1 ec2-user ec2-user 16 Nov 20 13:35 index.html
+ pwd
/var/lib/jenkins/workspace/testjob1
+ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 0e:e6:f5:0a:bc:94 brd ff:ff:ff:ff:ff:ff
    inet 172.31.47.177/20 brd 172.31.47.255 scope global dynamic eth0
       valid_lft 3445sec preferred_lft 3445sec
    inet6 fe80::ce6:f5ff:fe0a:bc94/64 scope link
       valid_lft forever preferred_lft forever
SSH: Connecting from host [ip-172-31-47-177.eu-west-3.compute.internal]
SSH: Connecting with configuration [websrv-1] ...
SSH: EXEC: completed after 201 ms
SSH: Disconnecting configuration [websrv-1] ...
SSH: Transferred 1 file(s)
Build step 'Send files or execute commands over SSH' changed build result to SUCCESS
Finished: SUCCESS
```

And open our websvr-1 in browser

← → C ⚠ Не защищено | 13.39.17.119

jenkins workflow

It works!))

Test it with git. For example used "[Conway's Game of Life in Javascript and canvas](#)" (Public access)

Create clone of testjob1 and change source on it:
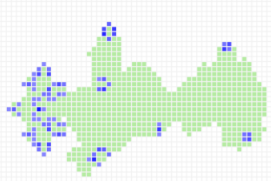
Conway's Game of Life

Running Information

Generation: 101 | Live cells: 83 | Step time: 1 / 0 (1 / 1) ms

Controls

Run Step Clear Export

Layout

Trail Grid Colors

Benchmarks (heavy patterns / experimental)

Benchmark #1   Benchmark #2

What is this?

This page contains a Javascript implementation of the well know John Conway's Game of Life. It uses the List Life algorithm (by Tony Finch) and a Canvas interface for drawing.

This is a work in progress and I will add some new features in the near future:

- Zoom controls
- Canvas navigation controls
- More patterns
- Proper benchmark support
- ...

Previous versions

- 1.0.0 - First public release. It uses the simplest algorithm possible and a table for drawing. Online here: pmav.eu.
- 2.0.0 - Private release, first implementation of the List Life algorithm.
- 3.1.1 - Current version, List Life and Canvas.

License

This work is licenced under a MIT license (except any 3rd party code).

3rd Party

Powered by CSS Browser Selector and JSON Sans Eval.

Links

Some links about Conway's Game of Life:

- Tony Finch's List Life Algorithm
- Golly - Application for exploring Conway's Game of Life
- Conway's Game of Life @ Wikipedia
- Life (B3/S23) @ Fano
- Life Page by Paul Callahan
- Game of Life Status page
- Life Lexicon Home Page

Branch name origin/master Build # 9 Sun Nov 20 16:47:24 UTC 2022

Create in my own git repository clone of this game:

```
Владимир@x270 MINGW64 ~
$ cd /d/MyProjects/git

Владимир@x270 MINGW64 /d/MyProjects/git
$ git clone git@github.com:pmav/game-of-life.git
Cloning into 'game-of-life'...
remote: Enumerating objects: 59, done.
remote: Total 59 (delta 0), reused 0 (delta 0), pack-reused 59
Receiving objects: 100% (59/59), 39.16 KiB | 471.00 KiB/s, done.
Resolving deltas: 100% (17/17), done.

Владимир@x270 MINGW64 /d/MyProjects/git
$ dir
L1-LinuxNet   TestCourse   game-of-life   myTest

Владимир@x270 MINGW64 /d/MyProjects/git
$ cd game-of-life

Владимир@x270 MINGW64 /d/MyProjects/git/game-of-life (master)
$ cd ..

Владимир@x270 MINGW64 /d/MyProjects/git
$ git clone git@github.com:WladimirRogovenko/DemoLifeGame.git
Cloning into 'DemoLifeGame'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

Владимир@x270 MINGW64 /d/MyProjects/git
$
```

```
Владимир@x270 MINGW64 /d/MyProjects/git
$ rm game-of-life/.git -r
rm: remove write-protected regular file 'game-of-life/.git/objects/pack/pack-4597419fc46f6785d5ec04a35b46ebaec93ec361.idx'? y
rm: remove write-protected regular file 'game-of-life/.git/objects/pack/pack-4597419fc46f6785d5ec04a35b46ebaec93ec361.pack'? y

Владимир@x270 MINGW64 /d/MyProjects/git
$

Владимир@x270 MINGW64 /d/MyProjects/git
$ mv game-of-life/* DemoLifeGame/

Владимир@x270 MINGW64 /d/MyProjects/git
$ cd DemoLifeGame

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$ git add *

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$ git commit -m "Copy all files from repo https://github.com/pmav/game-of-life"
[main cd7e721] Copy all files from repo https://github.com/pmav/game-of-life
 8 files changed, 1525 insertions(+), 2 deletions(-)
 create mode 100644 LICENSE
 create mode 100644 assets/css/style.css
 create mode 100644 assets/js/css-browser-selector.js
 create mode 100644 assets/js/game-of-life-v3.1.1.js
 create mode 100644 assets/js/json-sans-eval.js
 create mode 100644 assets/js/stats.min.js
 create mode 100644 index.html

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$ git push
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 4 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (13/13), 15.96 KiB | 817.00 KiB/s, done.
Total 13 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:WladimirRogovenko/DemoLifeGame.git
   732eebb..cd7e721  main -> main

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$
```

WladimirRogovenko / **DemoLifeGame** Private

Unwatch 1 ▾   Fork 0 ▾   Star 0 ▾

<> Code   ⊙ Issues   ⇄ Pull requests   ⊙ Actions   ⊞ Projects   ⊘ Security   ⸍ Insights   ⚙ Settings

main ▾   1 branch   0 tags   Go to file   Add file ▾   <> Code ▾

About

WladimirRogovenko Copy all files from repo https://github.com/pmav/game-of-life   cd7e721 3 minutes ago   ⊙ 2 commits

this for testing jenkins

assets	Copy all files from repo https://github.com/pmav/game-of-life	3 minutes ago
LICENSE	Copy all files from repo https://github.com/pmav/game-of-life	3 minutes ago
README.md	Copy all files from repo https://github.com/pmav/game-of-life	3 minutes ago
index.html	Copy all files from repo https://github.com/pmav/game-of-life	3 minutes ago

☐ Readme
⚖ MIT license
☆ 0 stars
⊙ 1 watching
⑂ 0 forks

Releases

≣ README.md

Create keys-pair for Jenkis-Git

```
drwxrwxr-x 3 ec2-user ec2-user    78 Nov 20 12:40 jenkins
drwx------ 3 ec2-user ec2-user    19 Nov 18 10:54 .local
drwx------ 2 ec2-user ec2-user    90 Nov 20 10:39 .ssh
-rw-r--r-- 1 root     root    166537 Nov 18 09:24 ud?
[ec2-user@ip-172-31-47-177 ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_rsa): jenkins-git_key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in jenkins-git_key.
Your public key has been saved in jenkins-git_key.pub.
The key fingerprint is:
SHA256:BJbKx4oxZyRcNOf2M90tefLQOWuJQbcYfHB0vP4c1TE ec2-user@ip-172-31-47-177.eu-west-3.compute.internal
The key's randomart image is:
+---[RSA 2048]----+
|  . o+ +.    ..o..|
|   o •=..    . o E.|
|    + oo .   + o =|
|   o =.oo . o 0 +o|
|    * o  S . 0 0 .|
|   . .    o   0 * |
|            . =.o|
|              . o|
|                 |
+----[SHA256]-----+
[ec2-user@ip-172-31-47-177 ~]$ ls -la
total 188
drwx------ 7 ec2-user ec2-user   214 Nov 20 17:27 .
drwxr-xr-x 3 root     root        22 Nov 18 08:37 ..
-rw------- 1 ec2-user ec2-user  3085 Nov 20 16:09 .bash_history
-rw-r--r-- 1 ec2-user ec2-user    18 Jul 15  2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user   193 Jul 15  2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user   231 Jul 15  2020 .bashrc
drwx------ 4 ec2-user root        35 Nov 18 10:54 .cache
drwx------ 3 ec2-user ec2-user    16 Nov 18 10:54 .config
drwxrwxr-x 3 ec2-user ec2-user    78 Nov 20 12:40 jenkins
-rw------- 1 ec2-user ec2-user  1675 Nov 20 17:27 jenkins-git_key
-rw-r--r-- 1 ec2-user ec2-user   434 Nov 20 17:27 jenkins-git_key.pub
drwx------ 3 ec2-user ec2-user    19 Nov 18 10:54 .local
drwx------ 2 ec2-user ec2-user    90 Nov 20 10:39 .ssh
-rw-r--r-- 1 root     root    166537 Nov 18 09:24 ud?
[ec2-user@ip-172-31-47-177 ~]$
```

Add public key to GitHub

Add private key to Jenkins

Jenkins Credentials Provider: Jenkins

Add Credentials

Domain

Global credentials (unrestricted)

Kind

SSH Username with private key

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

ID ?

jenkis-git-key

Description ?

jenkis-git-key

Username

WladimirRogovenko

☐ Treat username as secret ?

Private Key

⦿ Enter directly

Key

Enter New Secret Below

```
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEA0jYlTZyeEY6AN8qJBViVdLVwKqNBFJgP7Ok32yf5Irvgus1m
qksKEH28C5PVbn5SBGBaatJzQjctiZV3AxM2LYXXRTL5vQqbUhT59DJEMQ3LLRu/
```

Passphrase

Add   Cancel

Check **Git Host Key Verification Configuration**

Git Host Key Verification Configuration

Host Key Verification Strategy ?

Known hosts file

Save   Apply

Jenkins 2.378

Add Github.com to /var/lib/jenkins/.ssh/known_hosts

[ec2-user@ip-172-31-47-177 .ssh]$ sudo ssh-keyscan github.com >> /var/lib/jenkins/.ssh/known_hosts

Change owner

[ec2-user @ip-172-31-47-177 .ssh]# sudo chown jenkins:jenkins /var/lib/jenkins/.ssh/known_hosts

```
[ec2-user@ip-172-31-47-177 jenkins]$ sudo ls -la /var/lib/jenkins/.ssh
total 12
drwx------  2 jenkins jenkins   25 Nov 20 18:14 .
drwxr-xr-x 17 jenkins jenkins 4096 Nov 20 18:16 ..
-rw-r--r--  1 jenkins jenkins  830 Nov 20 18:16 known_hosts
[ec2-user@ip-172-31-47-177 jenkins]$
```

Restart Jenkins

[ec2-user @ip-172-31-47-177 .ssh]# sudo systemctl restart Jenkins

Next – create Job

Other settings same as previous testjob-git

Run our Job

Check site on websrv-1

Triggers (web hooks)

Create copy **testjob-git** in **test-git-my-repo** and check on 2 sittings:

In git project change index.html. Add "-2" to game name for example. Add to git, commit and push:


```
<!DOCTYPE html>
<html>
    <head>
        <script src="assets/js/json-sans-eval.js"></script>
        <script src="assets/js/css-browser-selector.js"></script>
        <script src="assets/js/stats.min.js"></script>
        <script src="assets/js/game-of-life-v3.1.1.js"></script>

        <link rel="stylesheet" type="text/css" href="assets/css/style.css" />
        <meta http-equiv="Content-type" content="text/html;charset=UTF-8" />

        <title>Conway's Game of Life - Javascript/Canvas Implementation</title>
    </head>
    <body>
        <div id="wrapper">

            <div id="title">
                <a href=".">Conway's Game of Life-2</a>
            </div>

            <canvas id="canvas" width="901" height="433"></canvas>
```
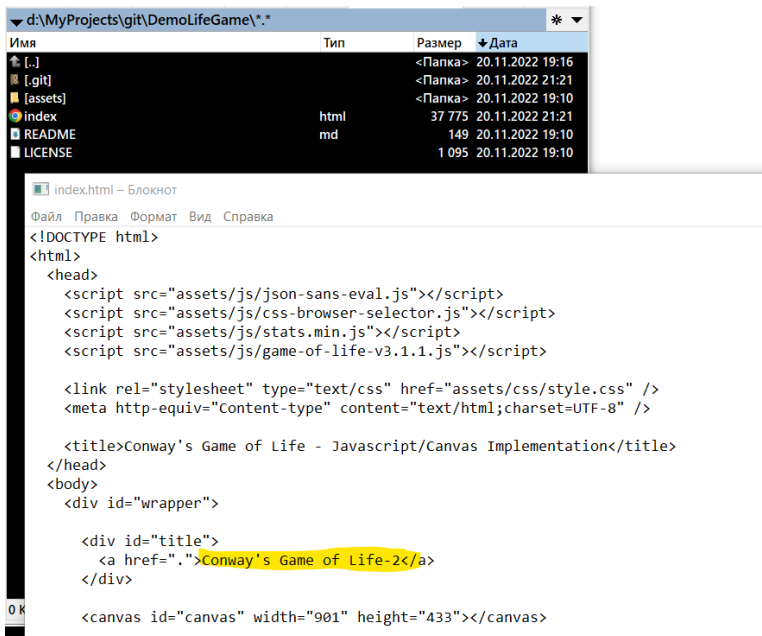
```
Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$ git add index.html

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$ git commit -m "test push -2"
[main 159815d] test push -2
 1 file changed, 1 insertion(+), 1 deletion(-)

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 307 bytes | 307.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:WladimirRogovenko/DemoLifeGame.git
   9d5227a..159815d  main -> main

Владимир@x270 MINGW64 /d/MyProjects/git/DemoLifeGame (main)
$
```

Since 10 seconds Jenkins deployed git project to websrv-1 automatically

Conway's Game of Life-2

- Tony Finch's List Life Algorithm
- Golly - Application for exploring Conway's Game of Life
- Conway's Game of Life @ Wikipedia
- Life (B3/S23) @ Fano
- Life Page by Paul Callahan
- Game of Life Status page
- Life Lexicon Home Page

Conway's Game of Life (B3/S23) v3.1.1 | **pmav.eu** | HTML 5 | 2008-2010

GitURL git@github.com:WladimirRogovenko/DemoLifeGame.git Branch name origin/main Build # 3 Sun Nov 20 19:21:47 UTC 2022

Add Jenkins Node

Create key pairs over ec2-key pairs with name "Jenkins-Nodes"

Create instance Node in same VPC and SubNet as Jenkins-Master:

Add to User Data next script:

```
#!/bin/bash
sudo yum update -y
sudo yum upgrade -y
sudo amazon-linux-extras install java-openjdk11 -y
sudo yum install git -y

#this command copied to /home/ec2-user/.shh/
sudo ssh-keyscan github.com >> /home/ec2-user/.ssh/known_hosts
```

Create new node

Search (CTRL+K)

1 Vladimir ⌄   log out

Dashboard > Manage Jenkins > Nodes >

New node

Node name

jenkins-node-1

Type

◉ Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

Create

Jenkins

Search (CTRL+K)

1 Vladimir ⌄   log out

Dashboard > Manage Jenkins > Nodes >

Name ?

jenkins-node-1

Description ?

Node for web projects

Number of executors ?

3

Remote root directory ?

/home/ec2-user

Labels ?

web dev java

Usage ?

Use this node as much as possible ⌄

Launch method ?

Save

Launch method ?

Launch agents via SSH

Host ?

172.31.47.143

Credentials ?

ec2-user (for jenkins nodes)

+ Add

Host Key Verification Strategy ?

Manually trusted key Verification Strategy

☐ Require manual verification of initial connection ?

Advanced...

Availability ?

Keep this agent online as much as possible

Node Properties

Save

Add new key

Jenkins Credentials Provider: Jenkins

Add Credentials

Domain

Global credentials (unrestricted)

Kind

SSH Username with private key

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

ID ?

jenkins-nodes-key

Description ?

for jenkins nodes

Username

Username

ec2-user

☐ Treat username as secret ?

Private Key

⦿ Enter directly

Key

Enter New Secret Below

```
NKWzAAGAiqWOEuYZdOBOEzXbEdyP+5JLM5h/5k/j4PJNaHBLVSKO/jj14+BHNBCSI
J10kbgeutVr-jOchzVyLPWpx1IEnm0bwxn117+IqZbF3V00tyQj18eNATGoOTO4A4
plAvaZ11r6uzAmIXLVMvmSYyqzPYU/hwNoGZdDOathpJKF3l0Ng=
-----END RSA PRIVATE KEY-----
```

Passphrase

Add   Cancel

Create copy of test-git-my-repo job with name test-git-my-repo-node-1

Run Job

Check our websrv: