Fitcrack

Distributed password cracking system

User manual

Radek Hranický, Adam Horák

Document for TARZAN project
Faculty of Information Technology, Brno University of Technology

Last modified: March 11, 2020

Contents

Getting Started with Fitcrack

Welcome to Fitcrack – The distributed password cracking system. This is your comprehensive and friendly user's guide. It focuses on the Webadmin app, which is the primary way of controlling the Fitcrack system. It was designed to be elegant and as user-friendly as possible for a password cracking control panel. Here's what you can expect to find in this documentaion:

- Getting familliar with the interface
- An overview of how Fitcrack works
- How to connect compute nodes
- How to create a cracking job
- What options are available for cracking
- How to work with jobs
- How to extend the system asset library
- How to manage users and resources

Before we can get started with all of that, however, you will want to have Fitcrack installed and ready to go. If you don't know what to do, please refer to the installation guide. After setting up, come back and we can get started.

Preflight Checks

After installing Fitcrack, you should have the system ready to use on your server. It is the main control center, housing the central database, communicating with all of the connected host nodes, and, crucially for us, running the Webadmin suite.

Now, depending on how you set the system up and whether you are exposing the web server to the internet, there will be different ways of connecting to Webadmin. In any case, you should be able to launch the app by visiting your server's domain via a web browser. When you do, you should be greeted by a login screen, just like below.

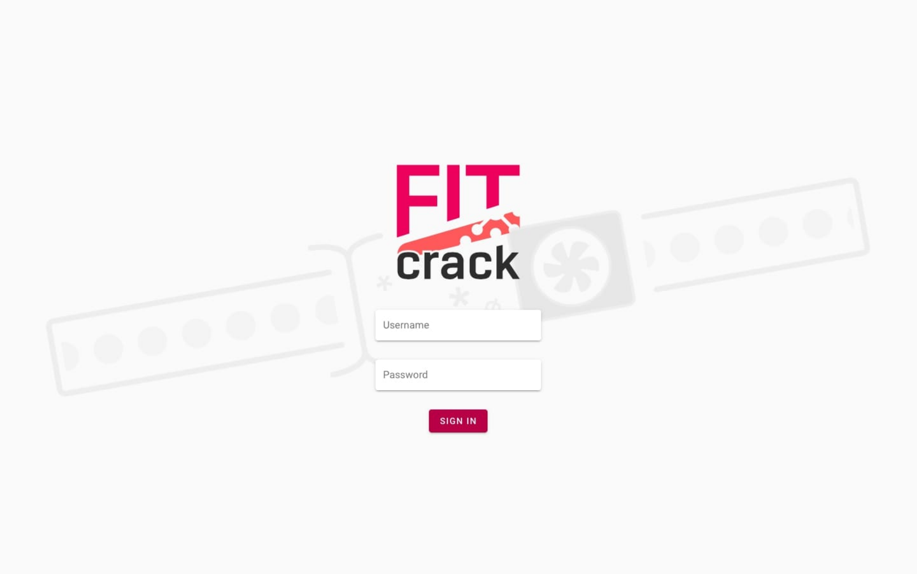


Figure 1: Fitcrack Login

First login

After a fresh install of Fitcrack, there is a default administrator user for Webadmin ready to go. To log in, use the following credentials: Username `fitcrack` with password `FITCRACK`.

It is a good idea to change the password after your first login. To do this, visit the *My Account* page from the main navigatoin bar on the left and use the *Change Password* box.

And that's it, you have successfully logged into Webadmin and are now ready to start using Fitcrack! Let's go take a look around the app next.

Navigating the App

Webadmin is a web application and you can use it on any device with a web browser that has access to your server. This means that the app will look different on various form factors, for example a phone.

After logging in, you will see a permanent app bar up at the top and a navigation menu on the left. These are always available regardless of current context.

> Note that the navigation may be hidden on narrower screens. To toggle it, use the hamburger menu icon on the far left of the app bar or swipe from the left edge on a touch screen.

App Bar and Notifications

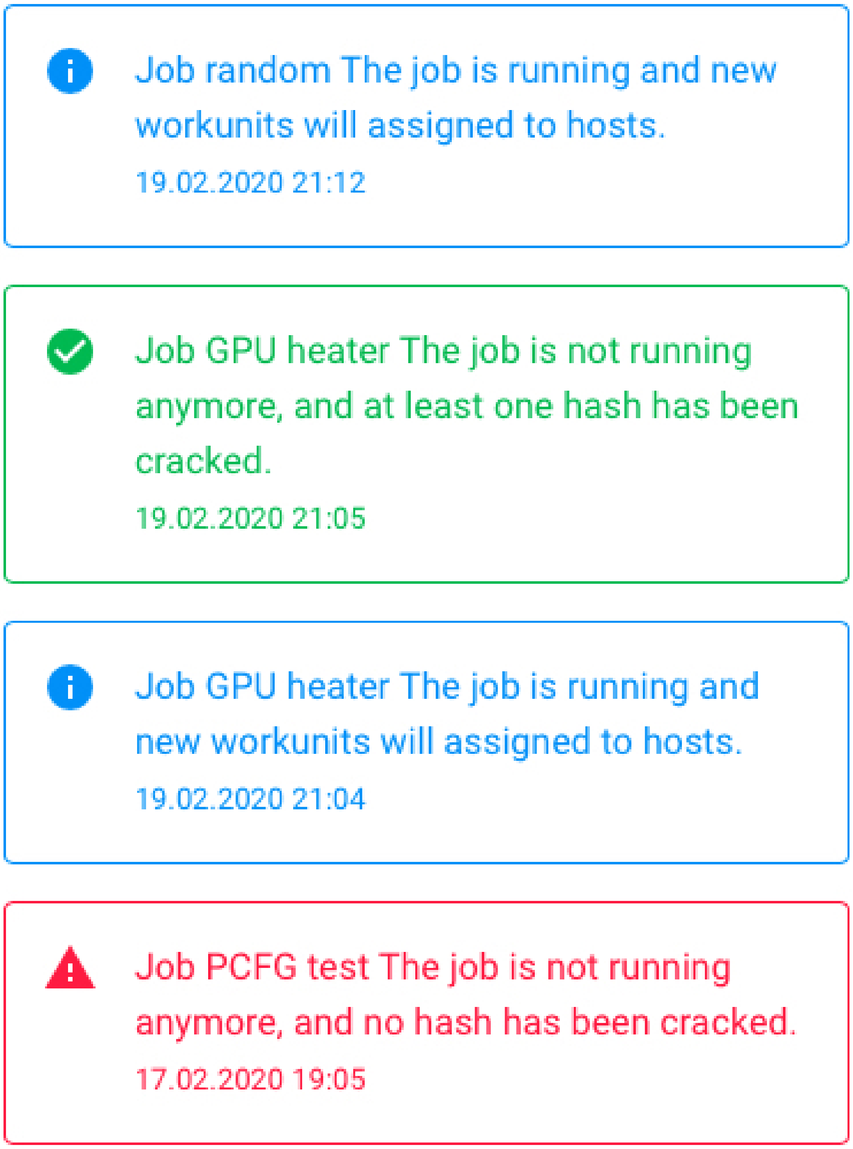
The app bar is always at the top and gives you access to the navigation, if it is hidden. On the right side, there is a button to open the notification drawer and a sign out button.

Figure 2: App Bar Actions

The notification drawer shows all of the notifications you received about job statuses and more. They are shown from newest to oldest.

Main Navigation

The main navigation links to every part of the system. You can find a very brief overview of each of them below.



Job random The job is running and new workunits will assigned to hosts.

19.02.2020 21:12

Job GPU heater The job is not running anymore, and at least one hash has been cracked.

19.02.2020 21:05

Job GPU heater The job is running and new workunits will assigned to hosts.

19.02.2020 21:04

Job PCFG test The job is not running anymore, and no hash has been cracked.

17.02.2020 19:05

Figure 3: Notification Drawer

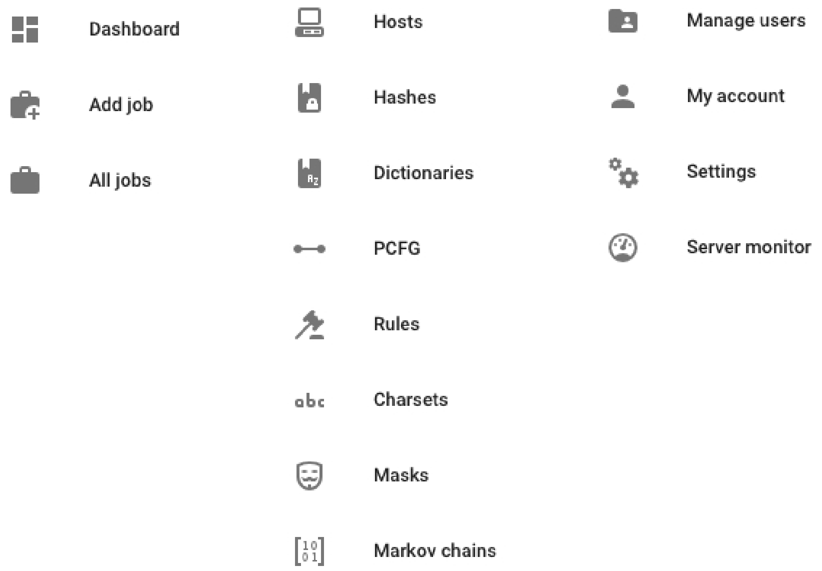


Figure 4: Navigation Bar

- **Dashboard** – The default view with an excerpt from critical parts of the system
- **Add Job** – The job creation form, see Creating Jobs
- **All Jobs** – Lists all created jobs, see Jobs List
- **Hosts** – Lists all host nodes connected to the system
- **Hashes** – Shows hashes known and stored by the system
- **Dictionaries, PCFG, Rules, Charsets, Masks and Markov Chains** – Shows the respective assets used in cracking and allows creating or uploading them
- **Manage Users** – Allows administrators to create and change user accounts, and to create or assign roles
- **My Account** – Shows the logged in user's info and allows them to change their password
- **Settings** – Shows various settings for Webadmin and also for the Fitcrack configuration
- **Server Monitor** – Shows server resource usage and service status

Host Nodes

Since Fitcrack is a distributed system, the server running Webadmin and the database is not itself doing the cracking (unless you want it to). The jobs are distributed between compute nodes (hosts) that you connect to the system using BOINC.

Hosts can only be added by attaching their BOINC client to the Fitcrack project on your server. They can also only be removed by detaching the project. In other words, you have no control over them in Webadmin.

When Fitcrack distributes workunits (parts of jobs), it takes into account the capabilities of each host. The workunits are tailor-made for that hardware's power.

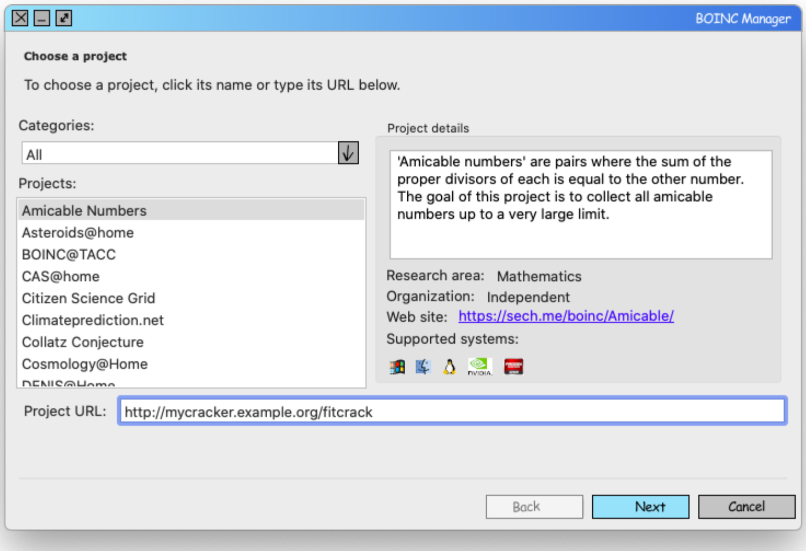
Connecting Hosts

To connect a machine to the system, it needs to run Windows or Linux. The BOINC client software is also required, as that is what Fitcrack uses to communicate with hosts. Install it from the official website or using a package manager on your system.

Using BOINC Manager

In a desktop environment, you can use the BOINC GUI. The screenshots below are actually platform-agnostic.

1. Click *Add Project* in BOINC (or, if in the advanced view, select *Tools > Add Project* from the menu bar)
2. Enter the URL of the server where your Fitcrack installation is running, with `/fitcrack` at the end
3. Create a new user (or use existing project credentials in later connections)
4. Finish adding the project, your host will soon appear in Webadmin



Choose a project

To choose a project, click its name or type its URL below.

Categories: All

Projects:

- Amicable Numbers
- Asteroids@home
- BOINC@TACC
- CAS@home
- Citizen Science Grid
- Climateprediction.net
- Collatz Conjecture
- Cosmology@Home
- DENIS@Home

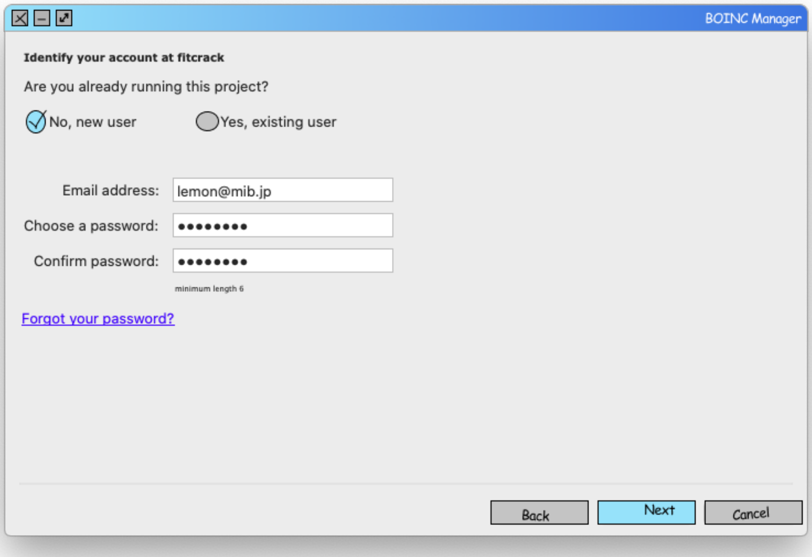
Project details

'Amicable numbers' are pairs where the sum of the proper divisors of each is equal to the other number. The goal of this project is to collect all amicable numbers up to a very large limit.

Research area: Mathematics
Organization: Independent
Web site: https://sech.me/boinc/Amicable/
Supported systems:

Project URL: http://mycracker.example.org/fitcrack

Figure 5: Adding a project in BOINC



Identify your account at fitcrack

Are you already running this project?

☑ No, new user ○ Yes, existing user

Email address: lemon@mib.jp
Choose a password: ••••••••
Confirm password: ••••••••
minimum length 6

Forgot your password?

Figure 6: Registering a user in BOINC

Note that BOINC Manager will open a webpage of the project to finish your registration. You should fill out your name and confirm, but you **do not have to search for or join a team!**

Using the command line

If you want to connect a host without a graphical display, for example a server you tunnel into via ssh, you can use the `boinccmd` utility.

1. **Get a key** by creating or looking up an account
    - To create an account, run `boinccmd --create_account <URL> <email> <password> <name>` (Mirroring the Manager example: `boinccmd --create_account http://mycracker.example.org/fitcrack lemon@mib.jp fufufufu Lemon`)
    - If you have an existing account, run `boinccmd --lookup_account <URL> <email> <password>` (With example credentials, this becomes `boinccmd --lookup_account http://mycracker.example.org/fitcrack lemon@mib.jp fufufufu`)
2. Copy the account key returned by the `create_account` or `lookup_account` command
3. Run `boinccmd --project_attach <URL> <account_key>` **with the key you copied** (Again, for the Manager example: `boinccmd --project_attach http://mycracker.example.org/fitcrack 3bc280b...`)
4. That's all, your host will soon appear in Webadmin

Creating a Job

This chapter will take you through the entire process of creating a cracking job in Webadmin.

In this section, we'll first take a look at this process as a whole. The following sections talk about available input options and attack modes in greater detail.

Getting Started with Jobs

When you navigate to the *Add Job* page, you will see a form split into four distinct steps:

- Input settings – preparing hashes to crack
- Attack settings – configuring an attack method
- Host assignment – planning distribution across compute nodes
- Additional settings – fine tuning and planning

There are two fields above the main form. The name field **is required**, as the job name will serve as an identifier when accessing it later. The template list is optional. Templates can serve as a starting point when creating a job, filling out parts of the form with previously saved values.

Let's take a look at each of the steps of the main form.

> Please note that fields marked with an asterisk are **required**!

Adding Input Hashes

In this first step, you input the hashes you want to try to crack. There are three ways to do so:

- Inputting the hashes manually
- Reading hashes from a file you upload
- Extracting from a protected file, like a PDF

The next section takes an in-depth look at each of them. Select a method using the toggle button **(1)** and enter your hashes in the text area **(2)** or upload

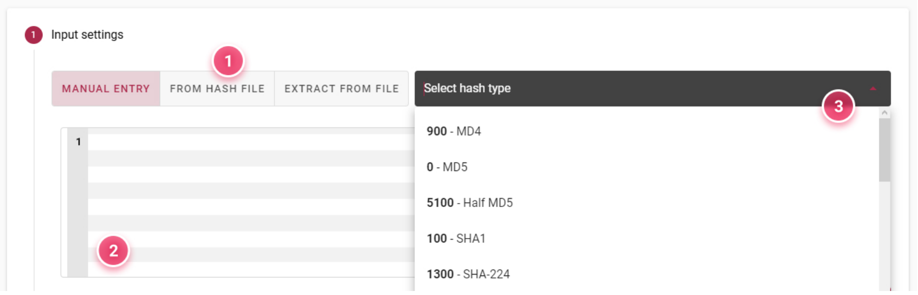
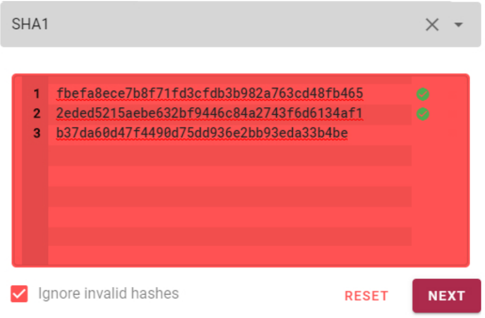


Figure 7: Input settings step

your files. Depending on the input method you chose, Fitcrack may be able to automatically detect the type of hash. If not, you have to select it manually from the dropdown **(3)**.

After setting the hash type, each hash is validated. **Green checkmarks** indicate the hash is valid for this type (algorithm). Hashes that have already been entered into the system and are stored in the cache will have a **yellow exclamation mark**. This is purely informational. Invalid hashes, on the other hand, should not be used in the configuration. If one or more hashes are invalid, the input field is marked **red** and you won't be able to create the job.



SHA1

```
fbefa8ece7b8f71fd3cfdb3b982a763cd48fb465
2eded5215aebe632bf9446c84a2743f6d6134af1
b37da60d47f4490d75dd936e2bb93eda33b4be
```

☑ Ignore invalid hashes

Figure 8: Invalid input settings

Note that you can override this using the *Ignore invalid hashes* checkbox that will appear. However, if you do this, there is no guarantee the job will work.

Configuring Attacks

In this step, you choose an attack mode and configure it to fit the job. There are a few different attack modes to choose from, each with its own specific setup. You can find all of them, along with a detailed description and examples, in the Attack modes section.

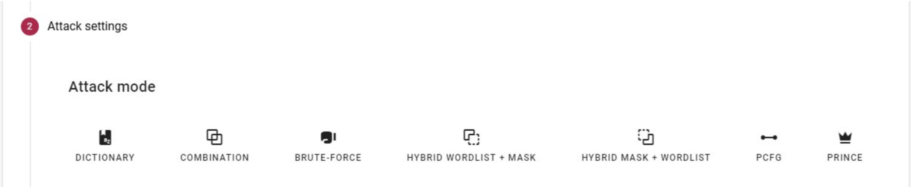
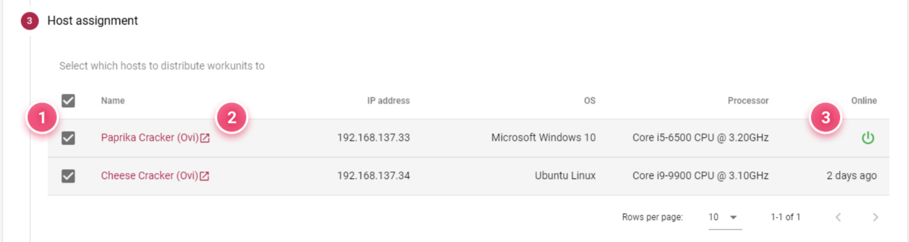


Figure 9: Attack modes

Once you've set up the attack using valid combination of options, you should be able to create the job, assuming you provided a name for it and valid input in the previous step. But let's not get ahead of ourselves, there are two more steps to look at.

Assigning Hosts



Select which hosts to distribute workunits to

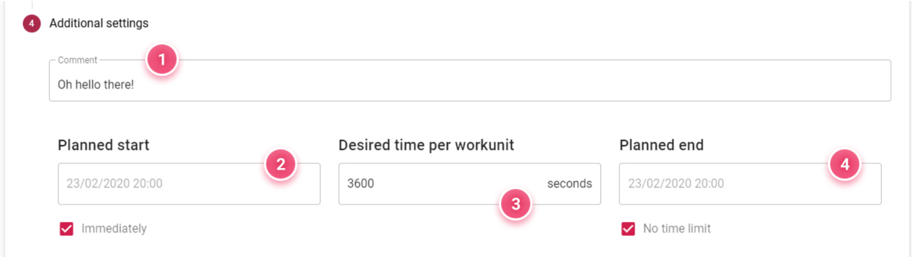
☑	Name	IP address	OS	Processor	Online
☑	Paprika Cracker (Ovi)	192.168.137.33	Microsoft Windows 10	Core i5-6500 CPU @ 3.20GHz	⏻
☑	Cheese Cracker (Ovi)	192.168.137.34	Ubuntu Linux	Core i9-9900 CPU @ 3.10GHz	2 days ago

Figure 10: Hosts table

In this step, you tap into the real potential of the Fitcrack system. The job we've been configuring until now will be distributed between compute nodes, called hosts, based on their measured computing power. You can select which of the hosts currently connected to the system will be participating in the cracking.

> Don't see any hosts in the table? See the guide in the Host nodes section of the introductory chapter.

The table lists all available hosts. By default, all hosts are selected. To select or deselect a host for taking part in this job, use the checkboxes in the table **(1)** or use the checkbox in the table header to toggle all. Clicking the name of the host **(2)** will take you to its detail page. Not all hosts will necessarily be online at the time of creating the job. To see if a host is online or how long ago it was last seen, check the *Online* column **(3)**. A green power icon indicates the host is available right now.



4 Additional settings

Comment: Oh hello there!

Planned start: 23/02/2020 20:00

☑ Immediately

Desired time per workunit: 3600 seconds

Planned end: 23/02/2020 20:00

☑ No time limit

Figure 11: Additional settings

Tuning the Details

This is the last step and it's very much optional. Here you can add a comment **(1)** to the job, which will be visible in the detailed job view. You can also set a time and date for the job to start **(2)** or leave the checkbox labeled *Immediately* checked to let the job start when you, well, start it. Simmilarily, you can set a time and date for the job to end **(4)** or not set any limitation.

The *Desired time per workunit* field **(3)** sets how long each workunit (part of the job received by a host) should run on the host node. The default value for this field can also be set on the *Settings* page.

Finishing up

We have now explored the entire job creation form. There are now two ways forward. You can create a template that will be available in the dropdown described earlier. Note that not all settings from the form are saved to templates, namely input hashes and host assignment. Upon choosing to make a template, you will be prompted for a name (identical to the job name by default). You can continue to work with the form after the template is saved.

The other thing to do is, of course, to create the job. Assuming your settings are valid, the *Create* (or *Save for later* in case no hosts are assigned) button will be enabled. After the job is created, you will be taken to the detail view, where you can review it and start it up.

Hash Input

When creating a job, you will have to provide the hash or hashes to crack. Fitcrack Webadmin offers three ways to do this.

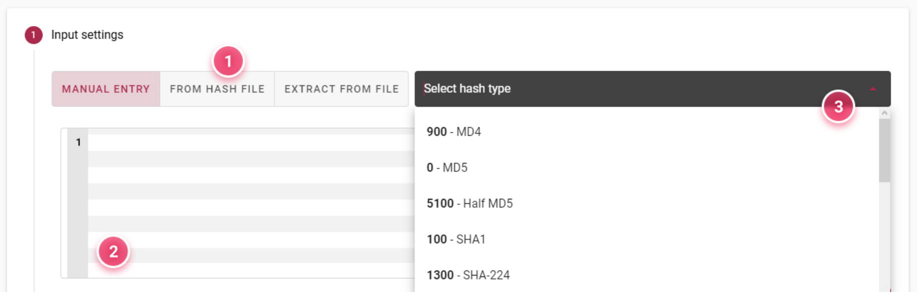


Figure 12: Input settings step

Select a method using the toggle button **(1)**. Depending on the input method you choose, Fitcrack may be able to automatically detect the type of hash after you enter it. If not, you have to select the type manually from the dropdown **(3)**.

Let's take a look at the available input methods:

Manual Entry

This method is selected by default. As the name suggests, you enter the hashes in text form in the input field **(2)** manually. Enter each hash on its own line.

This method cannot detect the hash type just from the text you enter, so you'll have to select the type from the hash type list **(3)**. After you do this, the hashes will be validated and valid lines will have a checkmark icon at the end.

For more on hash validation, see the Overview section in this chapter.

From Hash File

This method is simmilar to manual entry. Instead of inputting the hashes manually in a text field, however, you select a file to read. The file you select will not be uploaded to the server. Your browser will read the file contents and add the hashes to the input. From there, the proccess is identical as before, the hashes get validated after you select the hash type in the dropdown **(3)**.

The files you select should contain hashes on separate lines, just like when entering manually. One difference to the manual entry method is that you can select not only a text file, but a binary hash file too.

Extract from File

Using this method, you can extract hashes from encrypted files. An example of this might be a password protected PDF document or an archive, like zip or 7z. To do this, Webadmin uses a tool called XtoHashcat, that runs on the server; hence, the selected file will have to be uploaded to the server and processed there.

If the tool succeeds in extracting the hash, it will be added to the input. The hash type will also be selected automatically in most cases, and the hash will be validated immediately.

Attack Modes

Since Fitcrack uses Hashcat under the hood, it also provides the same attack modes. Attack modes refer to the different ways of cracking the hashes you provide as the job input.

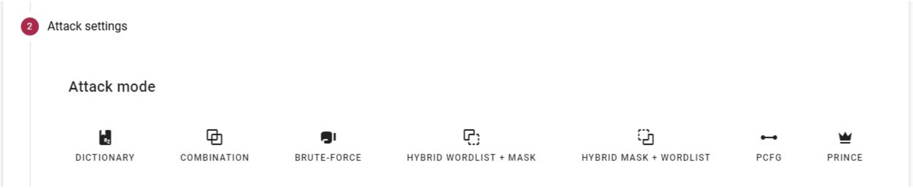


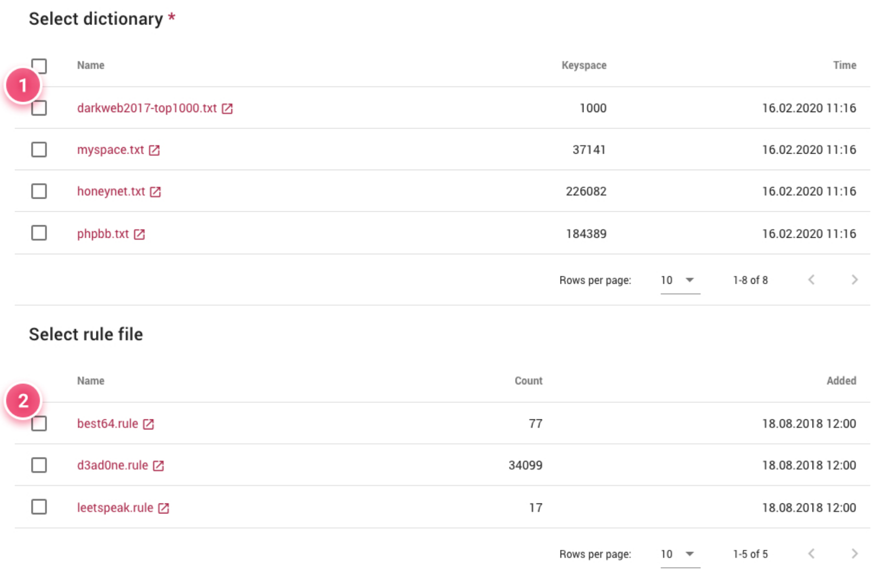
Figure 13: Attack modes

What attack mode to use depends on the job in question, but first, you need to get a feel for what each of them does. In this section, we'll take a closer look at each of the available modes.

Dictionary Attack

A dictionary attack will try every possible password from given password dictionaries. You can select one or more dictionaries you want to use using the table **(1)**. For each dictionary available, you can see the keyspace signifying the number of passwords which will be used. If you choose multiple dictionaries, the total keyspace of the job is a sum of keyspaces of all selected dictionaries.

It is also possible to select a file with password-mangling rules from the bottom table **(2)**. The number of rules in each file is shown in the *count* column. The rules enhance the repertoire of passwords, but they also increase the total keyspace of the job. This is because Fitcrack applies every rule from the rule file to each dictionary password. The total keyspace is calculated as the sum of dictionary keyspaces multiplied by the number of rules in the rule file.



Select dictionary *

☐	Name	Keyspace	Time
☐	darkweb2017-top1000.txt	1000	16.02.2020 11:16
☐	myspace.txt	37141	16.02.2020 11:16
☐	honeynet.txt	226082	16.02.2020 11:16
☐	phpbb.txt	184389	16.02.2020 11:16

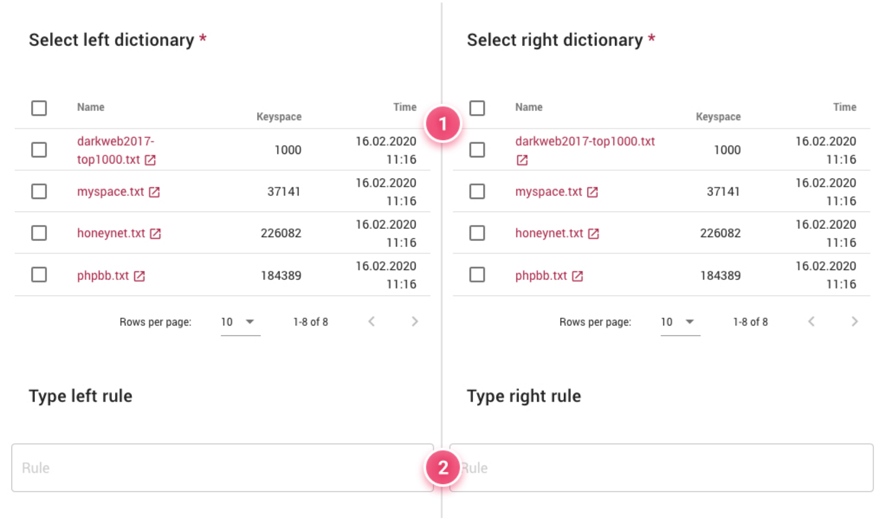
Rows per page: 10 1-8 of 8

Select rule file

	Name	Count	Added
☐	best64.rule	77	18.08.2018 12:00
☐	d3ad0ne.rule	34099	18.08.2018 12:00
☐	leetspeak.rule	17	18.08.2018 12:00

Rows per page: 10 1-5 of 5

Figure 14: Dictionary attack



Select left dictionary *

☐	Name	Keyspace	Time
☐	darkweb2017-top1000.txt	1000	16.02.2020 11:16
☐	myspace.txt	37141	16.02.2020 11:16
☐	honeynet.txt	226082	16.02.2020 11:16
☐	phpbb.txt	184389	16.02.2020 11:16

Rows per page: 10 1-8 of 8

Type left rule

Rule

Select right dictionary *

☐	Name	Keyspace	Time
☐	darkweb2017-top1000.txt	1000	16.02.2020 11:16
☐	myspace.txt	37141	16.02.2020 11:16
☐	honeynet.txt	226082	16.02.2020 11:16
☐	phpbb.txt	184389	16.02.2020 11:16

Rows per page: 10 1-8 of 8

Type right rule

Rule

Figure 15: Combination attack

Combination Attack

The combination attack is based on **combining passwords from two dictionaries** you select as left and right side **(1)**. Each password from the left dictionary is concatenated with every single password from the right dictionary. Such newly created passwords are then used for cracking.

It is also possible to define password-mangling rules for both sides **(2)**. The left and/or right rule will be applied to every password from the respective dictionary before the concatenation mentioned above happens. Unlike in a dictionary attack, with rules in a combination attack, there is only one rule per dictionary and it is always applied, hence the rules **do not increase the password keyspace**.

Brute Force Attack

The brute-force attack allows the user to define one or more **password masks** which define how a password may look like. Fitcrack then tries **every possible permutation of characters** in each mask.

Password mask

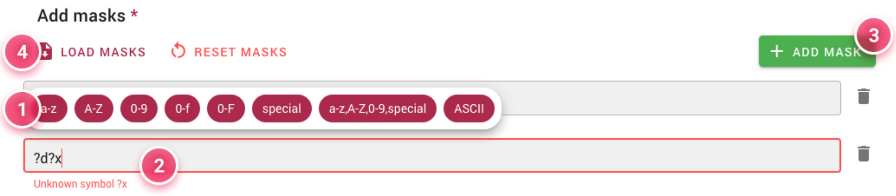
A mask is simply a template defining allowed characters on each position. It may contain either a concrete character, or a substitute symbol for a group of characters – e.g. **?l** for lowercase letter, or **?d** for digit.

For example, the mask `He?l?lo?d` represents all passwords between `Heaao0` and `Hezzo9`.

In masks, following basic substitute symbols are allowed:

- **?l** or **a-z** – lowercase Latin letters: `abcdefghijklmnopqrstuvwxyz`
- **?u** or **A-Z** – uppercase Latin letters: `ABCDEFGHIJKLMNOPQRSTUVWXYZ`
- **?d** or **0-9** – digits: `0123456789`
- **?s** or **special** – special ASCII characters including spaces, punctuation, etc.: `` (space)!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~ ``
- **?h** or **0-f** – characters representing hexadecimal digits with small letters: `0123456789abcdef`
- **?H** or **0-F** – characters representing hexadecimal digits with big letters: `0123456789ABCDEF`
- **?a** or **a-z,A-Z,0-9,special** – any character from **?l**, **?u**, **?d**, **?s**
- **?b** or **ASCII** – all ASCII characters starting from **0x00** (0) to **0xFF** (255)

You can create a mask by using a mask editor. For crafting masks, you can either use the popup bar with buttons **(1)** for inserting substitute symbols, or you can enter the mask by yourself in the input field **(2)** as text. The editor will turn red and show you if there is an error. You can also use multiple masks – if



?d?x

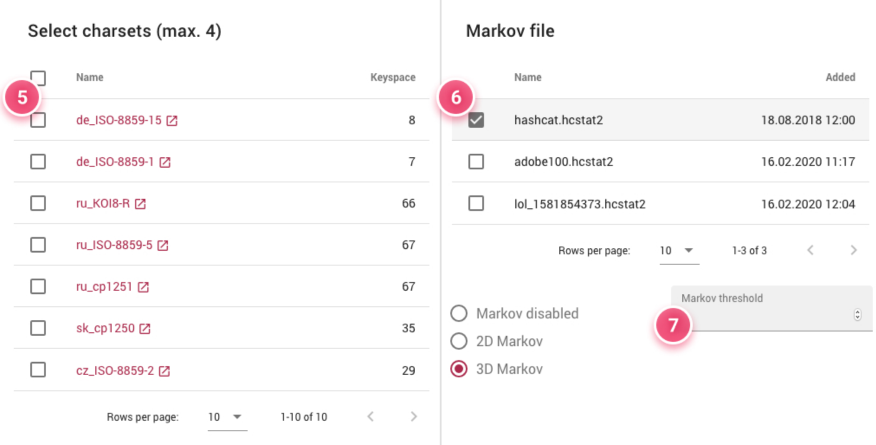
Unknown symbol ?x

Figure 16: Mask editors

you want to add a new mask, use the *Add masks* button **(3)**. Click the trash can icon next to the input field to delete that mask.

It is also possible to load masks. In Fitcrack, you do not have to enter masks manually every time you create a new job. You can have the masks stored in a mask set (.hcmask) file. Using the *Load masks* button **(4)**, you can choose a mask file, and the masks will be imported automatically. To remove all masks and start over, use the *Reset masks* button.

Custom character sets



☐	Name	Keyspace
☐	de_ISO-8859-15	8
☐	de_ISO-8859-1	7
☐	ru_KOI8-R	66
☐	ru_ISO-8859-5	67
☐	ru_cp1251	67
☐	sk_cp1250	35
☐	cz_ISO-8859-2	29

Rows per page: 10 1-10 of 10

	Name	Added
☑	hashcat.hcstat2	18.08.2018 12:00
☐	adobe100.hcstat2	16.02.2020 11:17
☐	lol_1581854373.hcstat2	16.02.2020 12:04

Rows per page: 10 1-3 of 3

○ Markov disabled
○ 2D Markov
● 3D Markov

Markov threshold

Figure 17: Brute force options

The basic set of substitute symbols can be enhanced by using custom character sets. In the charset table **(5)**, you can select up to four charset files with custom character sets. After adding a charset, a button for it will be available in the popup bar **(1)** and you will be able to use one or more of the following extra substitute symbols:

- **?1** – custom character set no. 1
- **?2** – custom character set no. 2

- **?3** – custom character set no. 3
- **?4** – custom character set no. 4

Markov chains

For generating passwords, the brute-force attack does not employ the traditional lexicographical order of characters by default (meaning **b** will be after **a**, etc.), but uses Markov chains instead. The order of generating password candidates is defined by a probability matrix or matrices in a Markov statistics file (with **.hcstat2** extension). For each brute-force attack, Fitcrack allows you to select the file with Markov statistics that will be used **(6)**.

You can select a mode or disable the use of markov chains to use lexicographical order, if you want. You can also define a threshold **(7)**, which will limit the keyspace of each character set to the number you choose.

Hybrid Attacks

Hybrid attacks combine the dictionary and brute-force approaches. The password candidates are crafted from two parts. One part is taken from a dictionary, just like in a combination attack. The other part is generated from a mask using the brute-force technique. Depending on which parts are made from dictionary and which from a mask, we can distinguish between two types of hybrid attacks:

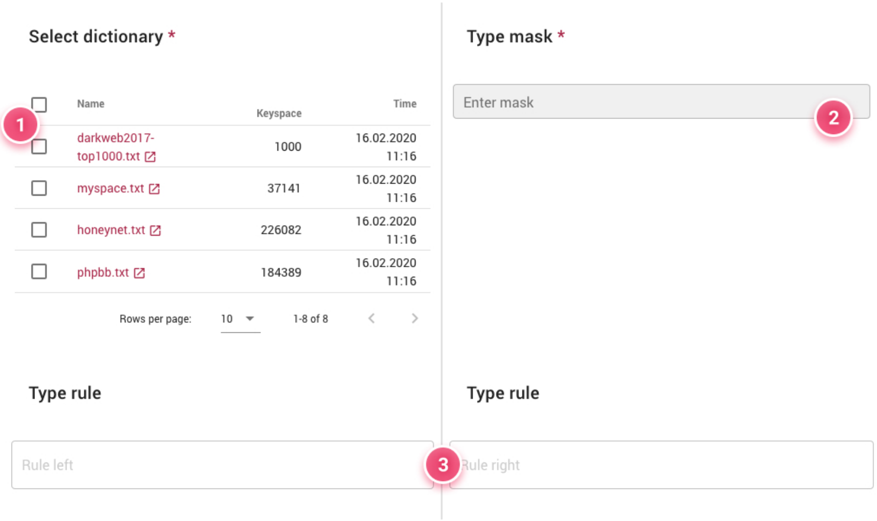
- Hybrid wordlist + mask – the left part is taken from a dictionary, the right part from a mask
- Hybrid mask + wordlist – the left part is generated from a mask, the right part is taken from a dictionary.

Select one or more dictionaries for the left or right part of the password **(1)**, and a mask for the other part **(2)**. As with the combination attack, you can define a password-mangling rule for both sides **(3)**.

PCFG Attack

PCFGs (Probabilistic Context Free Grammars) generate passwords using machine learning magic trained on password sets.

Select a grammar from the list **(1)** and optionally limit the keyspace below **(2)**. You can select a ruleset **(3)** to be used on the password candidates. These **do not, however, affect the keyspace**.



Select dictionary *

□	Name	Keyspace	Time
□	darkweb2017-top1000.txt	1000	16.02.2020 11:16
□	myspace.txt	37141	16.02.2020 11:16
□	honeynet.txt	226082	16.02.2020 11:16
□	phpbb.txt	184389	16.02.2020 11:16

Rows per page: 10 1-8 of 8

Type mask *

Enter mask

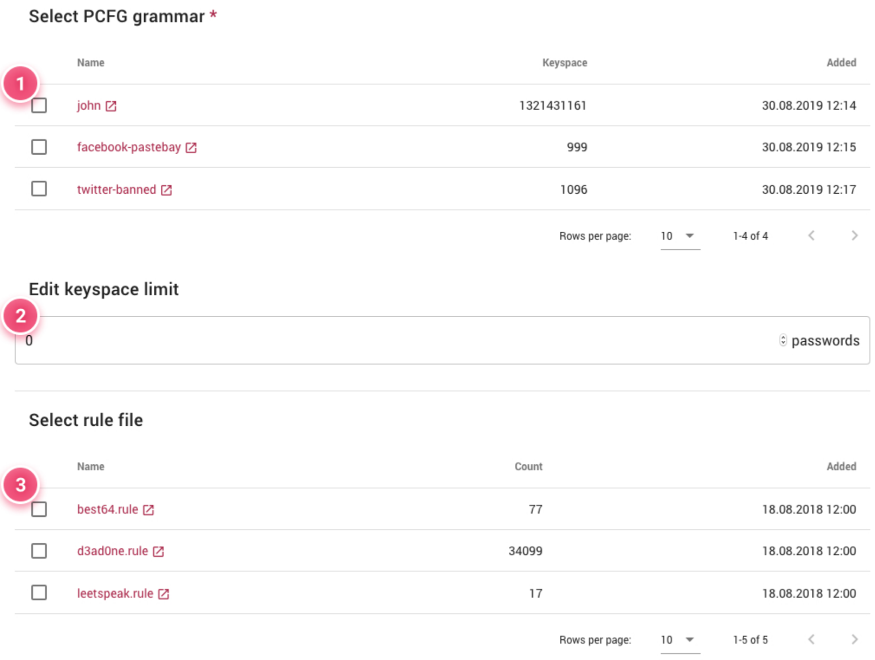
Type rule

Rule left

Type rule

Rule right

Figure 18: Hybrid attack



Select PCFG grammar *

	Name	Keyspace	Added
□	john	1321431161	30.08.2019 12:14
□	facebook-pastebay	999	30.08.2019 12:15
□	twitter-banned	1096	30.08.2019 12:17

Rows per page: 10 1-4 of 4

Edit keyspace limit

0 passwords

Select rule file

	Name	Count	Added
□	best64.rule	77	18.08.2018 12:00
□	d3ad0ne.rule	34099	18.08.2018 12:00
□	leetspeak.rule	17	18.08.2018 12:00

Rows per page: 10 1-5 of 5

Figure 19: PCFG attack

Jobs List

To see the jobs list, navigate to the *All Jobs* page via the main navigation bar. From here, you can also get to the individual jobs' detail views.

The jobs list shows all the jobs currently in the system in a paginated table, and allows you to search and filter them. Jobs can also be hidden from the main listing and are then shown separately.

Search and Filter

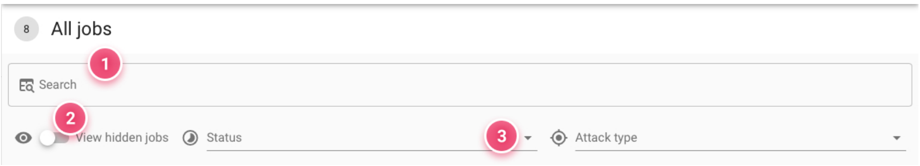


Figure 20: Search and filter panel

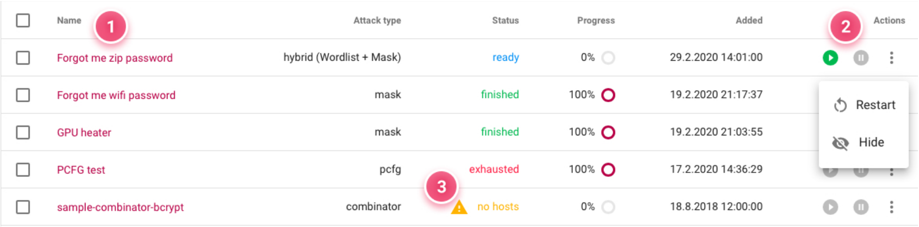
At the top of the list page is a search bar **(1)**, a toggle to switch between the default listing and the hidden one **(2)**, and filtering controls **(3)**.

You can search for a job by its name. To filter the list, you can choose a job status and/or an attack mode as a constraint.

Listed Jobs

Jobs satisfying the selected criteria are shown in a table view below the controls. The table is paginated and only show a selected number of rows at a time. Using some of the shown table headers **(1)**, you can also sort the table by their respective columns.

The table shows the job name, which is also a link to its detailed view, along some of the most useful stats to know at a glance, such as current state and progress.



□	Name	Attack type	Status	Progress	Added	Actions
□	Forgot me zip password	hybrid (Wordlist + Mask)	ready	0% ○	29.2.2020 14:01:00	▶ ⏸ ⋮
□	Forgot me wifi password	mask	finished	100% ○	19.2.2020 21:17:37	
□	GPU heater	mask	finished	100% ○	19.2.2020 21:03:55	
□	PCFG test	pcfg	exhausted	100% ○	17.2.2020 14:36:29	▶ ⏸ ⋮
□	sample-combinator-bcrypt	combinator	⚠ no hosts	0% ○	18.8.2018 12:00:00	▶ ⏸ ⋮

Figure 21: Jobs table

Move your cursor over the status text to see a brief description.

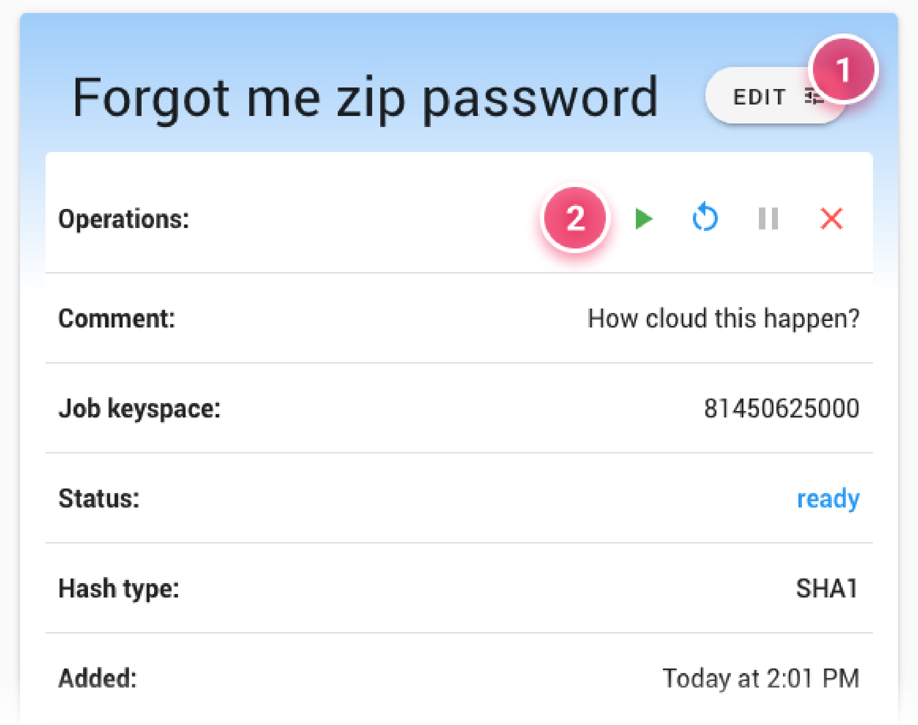
At the end of each row, there are quick actions **(2)** you can use to control the job right from the listing. Some options may be tucked away in the three dot menu.

From the list, you can also quickly see which jobs are missing required options, such as assigned hosts. This is represented by a warning icon **(3)** in the status column.

Job Detail

The job detail view is where you find everything there is to know about a specific job. From its setup to the passwords found, you can see statistics, breakdowns, logs from participating hosts and much more.

Main Info Panel



Forgot me zip password

Operations: ▶ ↻ ❚❚ ✕

Comment: How cloud this happen?

Job keyspace: 81450625000

Status: ready

Hash type: SHA1

Added: Today at 2:01 PM

Figure 22: Job detail panel

The main panel shows most of the information on the job setup, current state, progress and more. It is also wher you'll find the main controls **(2)** for starting, stopping and other operations for the job.

> For more on these controls, see Controlling Jobs.

The top area where the name is glows different colors depending on the current state of the job, so that you can tell what's happening at a glance. Here you can also open the edit dialog **(1)**, which allows you to change the job's name, description (comment) and reschedule its start and end times.

Attack Info

The attack info panel varies based on the attack mode used. It shows you how the attack was set up, for example what dictionaries are being used for a dictionary attack.

Hosts and Hashes

As their names suggest, these panels show you the hosts participating in the cracking and the hashes that are being cracked. When a password is found, you'll see it next to its hash.

At the bottom of the hosts panel, you can open the host assignment dialog and make changes.

Stat Graphs

Various statistics are aggregated while the job is running. These are then displayed in the form of different kinds of graphs. You can see how the job has progressed in time, for example, or see how the workunits were distributed between different host nodes.

Workunits and Logs

At the bottom of the page is a workunit table. It shows an overview of all the workunits in a job as a bar graphic, with a thorough breakdown in a table below, including complete logs for each workunit's run.

Controlling Jobs

During the lifecycle of a job in the system, it goes through different states. Depending on the current state, the job can be controlled in different ways.

When a job is created, it starts off in a ready state, not doing anything yet. It can be started or modified, and later in the lifecycle, it can be stopped, restarted or, should the need arise, purged.

Starting and Restarting

Jobs that are ready can be started. Jobs that are no longer running, be it after successful finish or in an error state, can be restarted and will run again.

Before a job can start, it has to have hosts assigned. If it doesn't, you will see a yellow warning and a button to assign the hosts instead of the typical job controls.

Stopping and Purging

You can stop a job prematurely or purge (kill) it if there are problems with it. As an example, some jobs may grind to a halt and get stuck in a finishing state. Purging such job will stop all participating hosts from working on it and will return the job to a clean ready state.

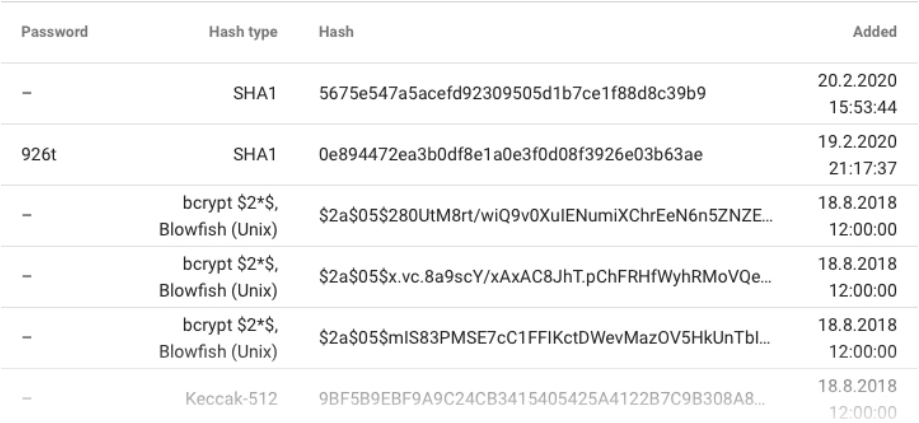
Reassigning Hosts

You may want to assign new hosts to a job or remove already assigned ones. This can be helpful when you want to add hosts that joined the system after the job was created, for example.

If you want to change the host assignment, you can do so from the Hosts panel in the detail view. The host assignment dialog updates automatically.

Hash Cache

The *Hashes* page lists all hashes currently stored in the hash cache. The system stores every hash that was entered, along with their passwords, if they were found.



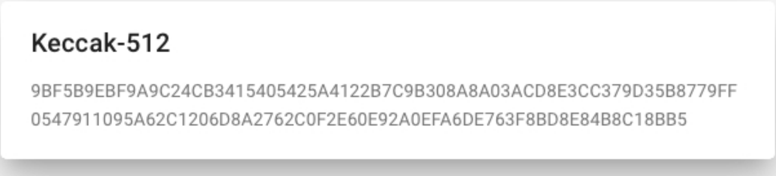
Password	Hash type	Hash	Added
–	SHA1	5675e547a5acefd92309505d1b7ce1f88d8c39b9	20.2.2020 15:53:44
926t	SHA1	0e894472ea3b0df8e1a0e3f0d08f3926e03b63ae	19.2.2020 21:17:37
–	bcrypt $2*$, Blowfish (Unix)	$2a$05$280UtM8rt/wiQ9v0XuIENumiXChrEeN6n5ZNZE...	18.8.2018 12:00:00
–	bcrypt $2*$, Blowfish (Unix)	$2a$05$x.vc.8a9scY/xAxAC8JhT.pChFRHfWyhRMoVQe...	18.8.2018 12:00:00
–	bcrypt $2*$, Blowfish (Unix)	$2a$05$mIS83PMSE7cC1FFIKctDWevMazOV5HkUnTbl...	18.8.2018 12:00:00
–	Keccak-512	9BF5B9EBF9A9C24CB3415405425A4122B7C9B308A8...	18.8.2018 12:00:00

Figure 23: Hash Cache

This is why you can sometimes see the message *Hash already in hash cache* when entering hashes into a cracking job input. You can still create a job with such hashes, but you may also find them in the cache and skip the cracking altogether.

If a hash doesn't fit the table, it will be truncated with ellipsis. You can see all of it by clicking on it, which will pop up a box with the full hash.

The Fitcrack installation also comes with a set of preloaded hashes. These function as a default rainbow table.



Keccak-512

9BF5B9EBF9A9C24CB3415405425A4122B7C9B308A8A03ACD8E3CC379D35B8779FF
0547911095A62C1206D8A2762C0F2E60E92A0EFA6DE763F8BD8E84B8C18BB5

Figure 24: Full Hash Popup

Cracking Assets

There are different kinds of assets used by the various attack modes available in Fitcrack, such as dictionaries, rulesets, and more. Many of these can be managed on their respective pages in Webadmin. You can edit or remove the ones shipping with Fitcrack, and you can, of course, upload new ones, extending the system's capabilities.

Dictionaries

The Dictionaries page allows you to manage password dictionaries. A dictionary is simply a text file (with a .txt extension) containing different password candidates where each password is stored on a separate line. Dictionaries are used for dictionary attacks, a combination attacks, and hybrid attacks.

Each dictionary can be opened and searched by clicking its name. You can download or delete dictionaries. To add a new dictionary, you can directly upload the text file, or pick one you uploaded to the server's import directory.

Dictionaries can also be sorted on creation. This will sort the lines by length, which can lead to significant performance boosts when cracking certain hashes, for example a 7zip archive.

PCFGs

PCFGs generate passwords using machine learning magic trained on password sets.

To define a new one, you can upload a zip archive directly or select a dictionary to create it from.

Rulesets

Rules define various modifications of password candidates. Such alterations include replacing and swapping of characters and substrings, password truncation, etc. Fitcrack uses Hashcat, which uses its own rule engine. Supported rules can be found in the Hashcat docs.

While the combinator and hybrid attacks allow the use of only one rule definition for each part, the dictionary and PCFG attacks can use a ruleset file.

A ruleset file is a text file which can contain one or more password-mangling rules on each line. The rules are all applied to every candidate password in the dictionary, meaning the password count will be a product of the dictionary size and the rule count.

Character Sets

In password masks used within brute-force or hybrid attacks, you can use the substitute symbols `?1` to `?4` for custom characters. The custom user-defined character sets are stored within a charset file with a .txt, .hcchr or .charset extension, which may contain both ASCII and non-ASCII characters.

Charsets can be edited by clicking the *Edit* button in their detail view.

Masks

Mask files contain a set of masks on separate lines. These can be loaded into a brute force attack when creating a job.

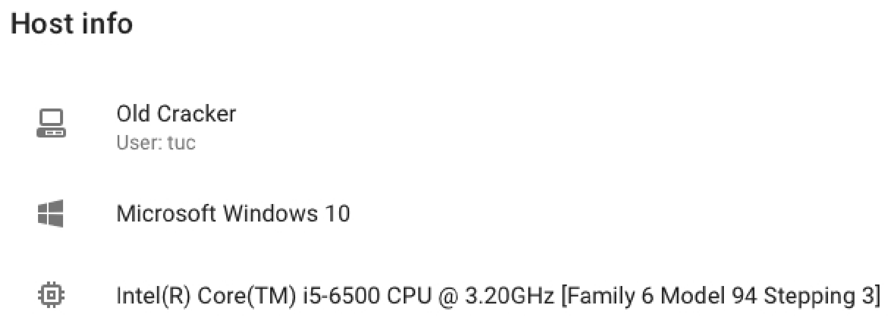
Markov Chains

Markov chains are used in brute borce attacks. They define what characters from a set go after another.

To define a new one, you can upload a hcstat2 file directly or select a dictionary to create it from.

Host Management

The hosts page lists all hosts currently attached to the system's BOINC server. You can see their name, basic specs and whether they are online or, if not, when they were last seen.



Host info

Old Cracker
User: tuc

Microsoft Windows 10

Intel(R) Core(TM) i5-6500 CPU @ 3.20GHz [Family 6 Model 94 Stepping 3]

Figure 25: Host Info

Clicking on a host will show you more information about them, with a breakdown of what jobs they were taking part in below. The bottom table then shows workunits that were given to this host to work on.

> If you are looking for guidance on attaching or detaching hosts to/from the system, see the introduction to hosts.

Server Monitor

The server monitor page shows which Fitcrack and BOINC services are running on the server (the one running Webadmin as well) and displays resource usage as graphs.

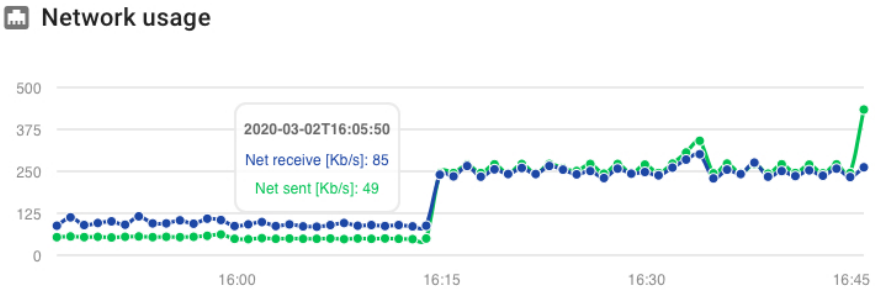


Figure 26: Network Usage

You can view concrete values by hovering over any point on the sparkline. To change the time interval for which data is shown, use the radio buttons at the bottom of the page.