

This document describes the caching system of SlapOS which is used to distributed binaries and support buildout building system. It consists of an sha self certifying server and a directory server which maps any key to an sha.

`{table_of_content}`

Shacache.org

shacache.org returns self certified data through sha. Accessing the URL:

shacache.org/fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307

returns a file which sha512 is

fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307

Two implementations of shacache.org exist: a local cache with no credential handling and a distributed cache which require valid credentials to post a file (and possibly pay for the traffic).

Shadir.org

shadir.org provides a mapping between an arbitrary key such as **pypi-buildout-3.1.2** and a list of signed JSON structures which contain file metadata including sha values. Accessing the URL:

shadir.org/pypi-buildout-3.1.2

returns a text file consisting of a list of signed JSON structures:

```
{
"file": "buildout-3.1.2",
"urlmd5": "39843943048",
"sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN X509 SIGNATURE-----
Version: 2.6.3a
Charset: noconv

iQCVAwUBMXVGMFla2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzrmGxV7xXzK0jus+6V
rML3TxQSwD= =T9Mc
-----END PGP SIGNATURE-----
{
"file": "buildout-3.1.2",
"urlmd5": "39843943048",
"sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN X509 SIGNATURE-----
Version: 2.6.3a
Charset: noconv

Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFla2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzrmGxV7xXzK0jus+6V
rML3TxQSwD= =T9Mc
```

Note: that a single JSON structure can be signed multiple times by different parties. The meaning of this signature is "party X trusts that the sha512 of buildout-3.1.2 is

fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307

Key Conflicts

From time to time different users of ShaDir will use the same key for files which sha512 and metadata is completely different. Or sometimes, the sha512 is the same but the metadata is different.

In this case, the list of signed JSON structure could provide different metadata with different signature.

Accessing

shadir.org/pypi-buildout-3.1.2

could return:

```
{
"file": "buildout-3.1.2",
"urlmd5": "39843943048",
"sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv
```

```
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
mZcD+XNnAaYcQfZ8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzmGxV7xXzK0jus+6V
rML3TxQSwDA= =T9Mc
-----END PGP SIGNATURE-----
{
  "file": "buildout-3.1.2",
  "urlmd5": "39843943048",
  "sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307,
  "distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv

Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYcQfZ8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzmGxV7xXzK0jus+6V
rML3TxQSwDA= =T9Mc
-----END PGP SIGNATURE-----
{
  "file": "buildout-3.1.2a",
  "urlmd5": "39843943018",
  "sha512": "dwd1ab7e49d05bedfd7ded135f0f96d99191570a0ABC8e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307,
  "distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv

Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYcQfZ8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzmGxV7xXzK0jus+6V
rML3TxQSwDA= =T9Mc
-----END PGP SIGNATURE-----
```

What is interesting in this case pypi-buildout-3.1.2 related to different entries in the directory with different sha512. One entry is signed twice.

Key conflicts as a way to implement Aging

By adding additional metadata to the JSON structure, it is possible to cache the different versions of the same cache entry. For example, the different checkouts of a given repository.

Accessing

shadir.org/pypi-buildout-3.1.2

would thus return:

```
{
  "file": "buildout-3.1.2",
  "urlmd5": "39843943048",
  "creation_date": 2011-04-01 10:10,
  "expiration_date": 2011-04-04 10:10,
  "sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307,
  "distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv
```

```
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
mZcD+XNnAaYcQfZ8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzmGxV7xXzK0jus+6V
rML3TxQSwDA= =T9Mc
-----END PGP SIGNATURE-----
{
  "file": "buildout-3.1.2",
  "urlmd5": "39843943048",
  "creation_date": 2011-05-01 10:10,
  "expiration_date": 2011-05-04 10:10,
  "sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307,
  "distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv
```

```
Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYcQfZ8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzmGxV7xXzK0jus+6V
rML3TxQSwDA= =T9Mc
-----END PGP SIGNATURE-----
{
  "file": "buildout-3.1.2a",
  "urlmd5": "39843943018",
  "creation_date": 2011-05-02 10:10,
  "expiration_date": 2011-05-06 10:10,
  "sha512": "dwd1ab7e49d05bedfd7ded135f0f96d99191570a0ABC8e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307,
  "distribution": "pypi"
}
```

-----BEGIN PGP SIGNATURE-----

Version: 2.6.3a
Charset: noconv

```
ZZJ9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzrmGxV7xXzK0jus+6V
rML3TxQSwdA= =T9Mc
-----END PGP SIGNATURE-----
```

This shows that by introducing `creation_date` and `expiration_date`, it is possible to store multiple entries for the same key in shadir. This allows to keep a history of cached files.

For example:

shadir.org/cli-39843943048

will return the list of cached files for the command line (cli) which md5 is 39843943048. This is very useful to cache certain keys which are by definition mutable, such as source code repository checkout.

```
{
"file": "svn co http://svn.erp5.org/public/erp5/trunk",
"urlmd5": "39843943048",
"creation_date": 2011-05-01 10:10,
"expiration_date": 2011-05-04 10:10,
"sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv

ZZJ9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzrmGxV7xXzK0jus+6V
rML3TxQSwdA= =T9Mc
-----END PGP SIGNATURE-----
{
"file": "svn co http://svn.erp5.org/public/erp5/trunk",
"urlmd5": "39843943018",
"creation_date": 2011-05-02 10:10,
"expiration_date": 2011-05-06 10:10,
"sha512": "dwd1ab7e49d05bedfd7ded135f0f96d99191570a0ABC8e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv

ZZJ9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzrmGxV7xXzK0jus+6V
rML3TxQSwdA= =T9Mc
-----BEGIN PGP SIGNATURE-----
```

Supporting Multiple Architectures

Thanks to key conflict, it is also possible to cache some file and take into account architecture. Accesing for example:

shadir.org/slapos-software-293982833

returns a text file:

```
{
"file": "erp5-5.1.2",
"urlmd5": "39843943048",
"creation_date": 2011-05-01 10:10,
"architecture": "x86-64 glibc 2.3",
"sha512": "fbd1ab7e49d05bedfd7ded135f0f96d99191570a02a98e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv

ZZJ9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFia2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzrmGxV7xXzK0jus+6V
rML3TxQSwdA= =T9Mc
-----END PGP SIGNATURE-----
{
"file": "erp5-5.1.2",
"urlmd5": "39843943018",
"creation_date": 2011-05-02 10:10,
"architecture": "x86-64 glibc 2.2",
"sha512": "dwd1ab7e49d05bedfd7ded135f0f96d99191570a0ABC8e84ea2feb9c19aed37c87d4aef3d305ba176bbc8971be9cc10e7d3b5b19c0eeb76ad04142ac4aaa1307",
"distribution": "pypi"
}
-----BEGIN PGP SIGNATURE-----
Version: 2.6.3a
Charset: noconv
```

Z2J9GVWD2zi4tG+s1+IWEY6Ae17kx925JKrzF4Ti2upAwTN2Pnb/x0G8WJQVKQzP
iQCVAwUBMXVGMFla2NdXHZJZAQFe4AQAz0FZrHdH8o+zklvcl/4ABg4gfE7cG0xE
mZcD+XNnAaYCqFz8iluAFVLchYeWj1Pqxxq0weGCtjQlrpzmGxV7xXzK0jus+6V
rML3TxQSwdA= =T9Mc
-----END PGP SIGNATURE-----

It is now clear that appropriate use of metadata helps resolving conflicts whenever the same key can be used for different software, for different cached versions of the software over time or for different target architectures.

Cache lookup process

The cache lookup process of buildout goes as follows:

1. If sha specified, use shacache
2. If no sha specified:
 1. If pypi specified use shadir/pypi-FILENAME
 2. If cli specified use shadir/cli-COMMANDMD5
 3. etc.
3. If shadir has no signed JSON structure with a key defined as relevant in the software configuration, then use original content**only**
4. If shadirs says content expired, try original content**first**
5. If original content fails or content not expired, use shacache

The cache lookup process for the full buildout, uses shadir with shadir.org/slapos-software-293982833 URLs

Signing Metadata

Some computers which download and build software can be granted the right to sign metadata JSON structures and attach it to a shadir key. For this purpose, an X509 signature certificate is placed on the machine. This X509 certificate is also registered on the shadir server. During the buildout process, if a given software is not available, it is built, then placed in an archive. The JSON structure is then generated and posted (POST) by REST interface to shadir.org together with the X509 signature.

Through this process, the number of signatures for a given key is extended. The same approach is implemented for binary cache.

Additional Notes

- We use X509 everywhere, not GPG or PGP.
- First implementation does not need to support signed JSON, but needs to support multiple JSON per key.

\${related_subject_list}