

GeniDesktop Parser API (PHP)

All API calls returns JSON objects of the form

```
{code: <value>,output : <value>}
```

If code = '1' in any of the API calls, then the value of the output would be a string with the appropriate error message. If code = '0', the value of the output would be the relevant JSON for the API call

1. getUserinfo

Params:

cert=<string>

passphrase=<string>

Return : Hash as value for output

```
{  
  uid : <username>,  
  userurn : <userurn>,  
  email : <email_address>,  
  certificate_issuer : <>,  
  user_crypt : <>  
}
```

The certificate issuer is the domain part of the user URN with '.' and ':' replaced with '_'
User crypt is a unique crypt identify the user for later use. All this info is extracted from the User' cert.

Sample:

```
{"output":{"uid":"demo","userurn":"urn:publicid:IDN+uky.emulab.net+user+demo","email":"demo@uky.emulab.net","certificate_issuer":"uky_emulab_net","user_crypt":"b8d91136asdadsdad26abb"},"code":0}
```

2. getSlicinfo

Params:

user_crypt=<string>

Return : Array of slice info as value for output

```
[[{sliceurn : <value>,slice_crypt : <string>,cmurn:<val>},{sliceurn : <value>,slice_crypt : <string>,cmurns : <val>}]]
```

This will provide a list of all the slices this users currently has along with a slice_crypt that uniquely identifies the slice.

Sample:

```
{"output":[{"sliceurn":"urn:publicid:IDN+uky.emulab.net+slice+hs1","crypt":"5SbmDBeeTXzSY","cmurns":null}], "code":0}
```

3. getSlicCred

Params:

user_crypt=<string>

slice_crypt=<string>

Return : Hash as value for output

```
{slicecred : <value>,uuid : <string>,expires:<val>}
```

This will provide the slice credential, the slice expiry and the uuid of the slice all obtained from the slice credential

Sample:

4. parseManifest

Params:

slice_crypt=<string>

api=< getLinkInfo | getNodeInfo >

force_refresh=<'1' or '0'> (optional, default is '0')

Return : Hash as value for output

JSON array of nodes or link as per request

This API call has more options to find information about a specific link or node also by providing an additional param. This call when called without a force_refresh will return a JSON from the saved manifest. If the manifest does not exist or the force_refresh is set, AMs will be polled to fetch the manifest and then the merged info will be returned in JSON Format

5. downloadManifest

Params:

slice_crypt=<string>

cmurn=< URN >

Return : a file (manifest in xml format)

6. getSliceAMs (To be implemented)

Params:

slice_crypt=<string>

cmurn=< URN > ? Still undecided

Return : none.