

NAME

rpc_gss_set_callback - Register a security context creation callback

LIBRARY

RPC GSS-API Authentication Library (librpcsec_gss, -lrpcsec_gss)

SYNOPSIS

```
#include <rpc/rpcsec_gss.h>
```

```
bool_t
```

```
(*callback)(struct svc_req *req, gss_cred_id_t deleg, gss_ctx_id_t gss_context, rpc_gss_lock_t *lock, void **cookie);
```

```
bool_t
```

```
rpc_gss_set_callback(rpc_gss_callback_t *cb);
```

DESCRIPTION

Register a function which will be called when new security contexts are created on a server. This function will be called on the first RPC request which uses that context and has the opportunity of rejecting the request (for instance after matching the request credentials to an access control list). To accept the new security context, the callback should return TRUE, otherwise FALSE. If the callback accepts a context, it becomes responsible for the lifetime of the delegated client credentials (if any).

It is also possible to 'lock' the values of service and quality of protection used by the context. If a context is locked, any subsequent requests which use different values for service and quality of protection will be rejected.

PARAMETERS

- | | |
|-------------|--|
| cb | A structure containing the RPC program and version for this callback and a function which will be called when new contexts are created for the given RPC program and version |
| req | The RPC request using the new context |
| deleg | GSS-API delegated credentials (if any) |
| gss_context | The GSS-API context |
| lock | A structure used to enforce a particular QOP and service. Set <i>lock->locked</i> to TRUE to lock the service and QOP values |

cookie The callback function may set **cookie* to any pointer sized value. This value can be accessed during the lifetime of the context via **rpc_gss_getcred()**.

RETURN VALUES

Returns TRUE if the callback was registered successfully or FALSE otherwise

SEE ALSO

gssapi(3), rpc(3), rpc_gss_getcred(3), rpcsec_gss(3)

HISTORY

The **rpc_gss_set_callback** function first appeared in FreeBSD 8.0.

AUTHORS

This manual page was written by Doug Rabson <dfr@FreeBSD.org>.

BUGS

There is no mechanism for informing a server when a security context has been deleted. This makes it difficult to allocate resources (e.g. to return via the callback's *cookie* argument).