
Subject: [PATCH 0/6] Lockd: service start cleanup patch set
Posted by [Stanislav Kinsbursky](#) on Wed, 25 Apr 2012 14:22:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

This patch set simplifies LockD start logic, makes code looks straight-forward and clear.

The following series consists of:

Stanislav Kinsbursky (6):

- LockD: pass service to per-net up and down functions
- LockD: use existent per-net data function on service creation
- LockD: service creation function introduced
- LockD: move global usage counter manipulation from error path
- LockD: service start function introduced
- LockD: add debug message to start and stop functions

fs/lockd/svc.c | 140 ++++++-----
1 files changed, 83 insertions(+), 57 deletions(-)

Subject: [PATCH 2/6] LockD: use existent per-net data function on service creation
Posted by [Stanislav Kinsbursky](#) on Wed, 25 Apr 2012 14:22:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

This patch also replaces svc_rpcb_setup() with svc_bind().

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

fs/lockd/svc.c | 23 +++++-----
1 files changed, 7 insertions(+), 16 deletions(-)

```
diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c
index 58ddc38..71c6c31 100644
--- a/fs/lockd/svc.c
+++ b/fs/lockd/svc.c
@@ -259,9 +259,9 @@ static int lockd_up_net(struct svc_serv *serv, struct net *net)
     if (ln->nlimsvc_users++)
         return 0;

- error = svc_rpcb_setup(serv, net);
+ error = svc_bind(serv, net);
     if (error)
- goto err_rpcb;
```

```

+ goto err_bind;

    error = make_socks(serv, net);
    if (error < 0)
@@ -270,7 +270,7 @@ static int lockd_up_net(struct svc_serv *serv, struct net *net)

err_socks:
    svc_rpcb_cleanup(serv, net);
-err_rpcb:
+err_bind:
    ln->nlimsvc_users--;
    return error;
}
@@ -298,7 +298,6 @@ int lockd_up(struct net *net)
{
    struct svc_serv *serv;
    int error = 0;
- struct lockd_net *ln = net_generic(net, lockd_net_id);

    mutex_lock(&nlimsvc_mutex);
/*
@@ -324,17 +323,9 @@ int lockd_up(struct net *net)
    goto out;
}

- error = svc_bind(serv, net);
- if (error < 0) {
-     printk(KERN_WARNING "lockd_up: bind service failed\n");
-     goto destroy_and_out;
- }
-
- ln->nlimsvc_users++;
-
- error = make_socks(serv, net);
+ error = lockd_up_net(serv, net);
    if (error < 0)
-     goto err_start;
+     goto err_net;

/*
 * Create the kernel thread and wait for it to start.
@@ -367,7 +358,7 @@ int lockd_up(struct net *net)
 * Note: svc_serv structures have an initial use count of 1,
 * so we exit through here on both success and failure.
 */
-destroy_and_out:
+err_net:
    svc_destroy(serv);

```

```
out:
  if (!error)
@@ -377,7 +368,7 @@ out:

err_start:
  lockd_down_net(serv, net);
- goto destroy_and_out;
+ goto err_net;
}
EXPORT_SYMBOL_GPL(lockd_up);
```

Subject: [PATCH 3/6] LockD: service creation function introduced
Posted by [Stanislav Kinsbursky](#) on Wed, 25 Apr 2012 14:22:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

This function creates service if it's not exist, or increase usage counter of the existent, and returns pointer to it.
Usage counter will be dropped by `svc_destroy()` later in `lockd_up()`.

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```
---
fs/lockd/svc.c | 38 ++++++-----
1 files changed, 27 insertions(+), 11 deletions(-)

diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c
index 71c6c31..ad11ea7 100644
--- a/fs/lockd/svc.c
+++ b/fs/lockd/svc.c
@@ -291,21 +291,20 @@ static void lockd_down_net(struct svc_serv *serv, struct net *net)
 }
 }

-/*
- * Bring up the lockd process if it's not already up.
- */
-int lockd_up(struct net *net)
+static struct svc_serv *lockd_create_svc(void)
 {
  struct svc_serv *serv;
- int error = 0;

- mutex_lock(&nlmsvc_mutex);
  /*
   * Check whether we're already up and running.
   */
  if (nlmsvc_rqst) {
```

```

- error = lockd_up_net(nlmsvc_rqst->rq_server, net);
- goto out;
+ /*
+ * Note: increase service usage, because later in case of error
+ * svc_destroy() will be called.
+ */
+ svc_get(nlmsvc_rqst->rq_server);
+ return nlmsvc_rqst->rq_server;
}

/*
@@ -316,11 +315,28 @@ int lockd_up(struct net *net)
    printk(KERN_WARNING
           "lockd_up: no pid, %d users??\n", nlmsvc_users);

- error = -ENOMEM;
  serv = svc_create(&nlmsvc_program, LOCKD_BUFSIZE, NULL);
  if (!serv) {
    printk(KERN_WARNING "lockd_up: create service failed\n");
- goto out;
+ return ERR_PTR(-ENOMEM);
+ }
+ return serv;
+}
+
+/*
+ * Bring up the lockd process if it's not already up.
+ */
+int lockd_up(struct net *net)
+{
+ struct svc_serv *serv;
+ int error = 0;
+
+ mutex_lock(&nlmsvc_mutex);
+
+ serv = lockd_create_svc();
+ if (IS_ERR(serv)) {
+ error = PTR_ERR(serv);
+ goto err_create;
+ }

  error = lockd_up_net(serv, net);
@@ -360,9 +376,9 @@ int lockd_up(struct net *net)
  /*
  err_net:
  svc_destroy(serv);
-out:
  if (!error)

```

```
nlmsvc_users++;
+err_create:
mutex_unlock(&nlmsvc_mutex);
return error;
```

Subject: [PATCH 4/6] LockD: move global usage counter manipulation from error path

Posted by [Stanislav Kinsbursky](#) on Wed, 25 Apr 2012 14:23:02 GMT

[View Forum Message](#) <> [Reply to Message](#)

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```
fs/lockd/svc.c | 5 +----
1 files changed, 2 insertions(+), 3 deletions(-)
```

```
diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c
index ad11ea7..53cd69e 100644
--- a/fs/lockd/svc.c
+++ b/fs/lockd/svc.c
@@ -329,7 +329,7 @@ static struct svc_serv *lockd_create_svc(void)
int lockd_up(struct net *net)
{
struct svc_serv *serv;
- int error = 0;
+ int error;

mutex_lock(&nlmsvc_mutex);

@@ -370,14 +370,13 @@ int lockd_up(struct net *net)
goto err_start;
}

+ nlmsvc_users++;
/*
* Note: svc_serv structures have an initial use count of 1,
* so we exit through here on both success and failure.
*/
err_net:
svc_destroy(serv);
- if (!error)
- nlmsvc_users++;
err_create:
mutex_unlock(&nlmsvc_mutex);
return error;
```

Subject: [PATCH 5/6] LockD: service start function introduced
Posted by [Stanislav Kinsbursky](#) on Wed, 25 Apr 2012 14:23:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

This is just a code move, which from my POW makes code looks better.
I.e. now on start we have 3 different stages:

- 1) Service creation.
- 2) Service per-net data allocation.
- 3) Service start.

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```
fs/lockd/svc.c | 67 ++++++++++++++++++++++++++++++++++++++-----  
1 files changed, 42 insertions(+), 25 deletions(-)
```

```
diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c  
index 53cd69e..b47bf77 100644
```

```
--- a/fs/lockd/svc.c  
+++ b/fs/lockd/svc.c  
@@ -291,6 +291,46 @@ static void lockd_down_net(struct svc_serv *serv, struct net *net)  
 }  
 }
```

```
+static int lockd_start_svc(struct svc_serv *serv)  
{  
+ int error;  
+  
+ if (nlmsvc_rqst)  
+ return 0;  
+  
+ /*  
+ * Create the kernel thread and wait for it to start.  
+ */  
+ nlmsvc_rqst = svc_prepare_thread(serv, &serv->sv_pools[0], NUMA_NO_NODE);  
+ if (IS_ERR(nlmsvc_rqst)) {  
+ error = PTR_ERR(nlmsvc_rqst);  
+ printk(KERN_WARNING  
+ "lockd_up: svc_rqst allocation failed, error=%d\n",  
+ error);  
+ goto out_rqst;  
+ }  
+  
+ svc_sock_update_bufs(serv);  
+ serv->sv_maxconn = nlm_max_connections;  
+  
+ nlmsvc_task = kthread_run(lockd, nlmsvc_rqst, serv->sv_name);  
+ if (IS_ERR(nlmsvc_task)) {  
+ error = PTR_ERR(nlmsvc_task);
```

```

+ printk(KERN_WARNING
+ "lockd_up: kthread_run failed, error=%d\n", error);
+ goto out_task;
+ }
+ dprintk("lockd_up: service started\n");
+ return 0;
+
+out_task:
+ svc_exit_thread(nlmsvc_rqst);
+ nlmsvc_task = NULL;
+out_rqst:
+ nlmsvc_rqst = NULL;
+ return error;
+}
+
static struct svc_serv *lockd_create_svc(void)
{
    struct svc_serv *serv;
@@ -343,32 +383,9 @@ int lockd_up(struct net *net)
    if (error < 0)
        goto err_net;

- /*
-  * Create the kernel thread and wait for it to start.
-  */
- nlmsvc_rqst = svc_prepare_thread(serv, &serv->sv_pools[0], NUMA_NO_NODE);
- if (IS_ERR(nlmsvc_rqst)) {
-     error = PTR_ERR(nlmsvc_rqst);
-     nlmsvc_rqst = NULL;
-     printk(KERN_WARNING
- "lockd_up: svc_rqst allocation failed, error=%d\n",
-     error);
-     goto err_start;
- }
-
- svc_sock_update_bufs(serv);
- serv->sv_maxconn = nlm_max_connections;
-
- nlmsvc_task = kthread_run(lockd, nlmsvc_rqst, serv->sv_name);
- if (IS_ERR(nlmsvc_task)) {
-     error = PTR_ERR(nlmsvc_task);
-     svc_exit_thread(nlmsvc_rqst);
-     nlmsvc_task = NULL;
-     nlmsvc_rqst = NULL;
-     printk(KERN_WARNING
- "lockd_up: kthread_run failed, error=%d\n", error);
+ error = lockd_start_svc(serv);
+ if (error < 0)

```

```
goto err_start;
- }

nlmsvc_users++;
/*
```

Subject: [PATCH 6/6] LockD: add debug message to start and stop functions
Posted by [Stanislav Kinsbursky](#) on Wed, 25 Apr 2012 14:23:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```
fs/lockd/svc.c | 5 +++++
1 files changed, 5 insertions(+), 0 deletions(-)
```

```
diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c
```

```
index b47bf77..80938fd 100644
```

```
--- a/fs/lockd/svc.c
```

```
+++ b/fs/lockd/svc.c
```

```
@@ -266,6 +266,7 @@ static int lockd_up_net(struct svc_serv *serv, struct net *net)
    error = make_socks(serv, net);
    if (error < 0)
        goto err_socks;
+ dprintk("lockd_up_net: per-net data created; net=%p\n", net);
    return 0;
```

```
err_socks:
```

```
@@ -283,6 +284,7 @@ static void lockd_down_net(struct svc_serv *serv, struct net *net)
    if (--nlmsvc_users == 0) {
        nlm_shutdown_hosts_net(net);
        svc_shutdown_net(serv, net);
+ dprintk("lockd_down_net: per-net data destroyed; net=%p\n", net);
    }
    else {
        printk(KERN_ERR "lockd_down_net: no users! task=%p, net=%p\n",
@@ -360,6 +362,7 @@ static struct svc_serv *lockd_create_svc(void)
    printk(KERN_WARNING "lockd_up: create service failed\n");
    return ERR_PTR(-ENOMEM);
}
+ dprintk("lockd_up: service created\n");
    return serv;
}
```

```
@@ -426,7 +429,9 @@ lockd_down(struct net *net)
    BUG();
}
```



```
kthread_stop(nlmsvc_task);
+ dprintk("lockd_down: service stopped\n");
  svc_exit_thread(nlmsvc_rqst);
+ dprintk("lockd_down: service destroyed\n");
  nlmsvc_task = NULL;
  nlmsvc_rqst = NULL;
out:
```

Subject: Re: [PATCH 0/6] Lockd: service start cleanup patch set
Posted by [bfields](#) on Wed, 09 May 2012 19:08:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wed, Apr 25, 2012 at 06:22:32PM +0400, Stanislav Kinsbursky wrote:
> This patch set simplifies LockD start logic, makes code looks straight-forward
> and clear.

I've applied and pushed these out.

I believe I now have everything from you except your latest grace-period handling rfc. Could you check and make sure that I'm not missing anything else, and that I haven't messed up anything else?

--b.

>
> The following series consists of:
>
> ---
>
> Stanislav Kinsbursky (6):
> LockD: pass service to per-net up and down functions
> LockD: use existent per-net data function on service creation
> LockD: service creation function introduced
> LockD: move global usage counter manipulation from error path
> LockD: service start function introduced
> LockD: add debug message to start and stop functions
>
>
> fs/lockd/svc.c | 140 ++++++-----
> 1 files changed, 83 insertions(+), 57 deletions(-)
>

Subject: Re: [PATCH 0/6] Lockd: service start cleanup patch set
Posted by [Stanislav Kinsbursky](#) on Wed, 09 May 2012 21:04:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

> On Wed, Apr 25, 2012 at 06:22:32PM +0400, Stanislav Kinsbursky wrote:
>> This patch set simplifies LockD start logic, makes code looks straight-forward
>> and clear.
> I've applied and pushed these out.
>
> I believe I now have everything from you except your latest grace-period
> handling rfc. Could you check and make sure that I'm not missing
> anything else, and that I haven't messed up anything else?

Sure, Bruce.
I'll check and reply in few days.

> --b.
>
>> The following series consists of:
>>
>> ---
>>
>> Stanislav Kinsbursky (6):
>> LockD: pass service to per-net up and down functions
>> LockD: use existent per-net data function on service creation
>> LockD: service creation function introduced
>> LockD: move global usage counter manipulation from error path
>> LockD: service start function introduced
>> LockD: add debug message to start and stop functions
>>
>>
>> fs/lockd/svc.c | 140 ++++++-----
>> 1 files changed, 83 insertions(+), 57 deletions(-)
>>

Subject: Re: [PATCH 0/6] Lockd: service start cleanup patch set
Posted by [Stanislav Kinsbursky](#) on Fri, 11 May 2012 11:04:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 09.05.2012 23:08, J. Bruce Fields wrote:
> On Wed, Apr 25, 2012 at 06:22:32PM +0400, Stanislav Kinsbursky wrote:
>> This patch set simplifies LockD start logic, makes code looks straight-forward
>> and clear.
>
> I've applied and pushed these out.
>
> I believe I now have everything from you except your latest grace-period
> handling rfc. Could you check and make sure that I'm not missing
> anything else, and that I haven't messed up anything else?
>

Looks like you took all except patches for Trond (NFS callback containerization).
Thanks.

```
> --b.  
>  
>>  
>> The following series consists of:  
>>  
>> ---  
>>  
>> Stanislav Kinsbursky (6):  
>>   LockD: pass service to per-net up and down functions  
>>   LockD: use existent per-net data function on service creation  
>>   LockD: service creation function introduced  
>>   LockD: move global usage counter manipulation from error path  
>>   LockD: service start function introduced  
>>   LockD: add debug message to start and stop functions  
>>  
>>  
>> fs/lockd/svc.c | 140 ++++++-----  
>> 1 files changed, 83 insertions(+), 57 deletions(-)  
>>
```

--
Best regards,
Stanislav Kinsbursky
