Suresec security advisory 4
6th July 2005
CVE ID: CAN-2005-1768

Linux kernel local root vulnerability (ia64 and amd64 ia32 compat.)

About the Linux kernel:
The Linux kernel is a widely used kernel which is unix based.

Vulnerability summary:
ia32 compatibility for the ia64 and amd64 platform has compatibility code for the execve() systemcall. A race condition was found within this systemcall. As will be shown by a code snippet this race condition leads to a buffer overflow.

```
static int nargs (unsigned int arg, char **ap) {
      unsigned int addr;
      int n=0, err;
      ...
      do {
            err = get_user(addr, (unsigned int *)A(arg));
            ...
            if (ap)
                  *ap++ = (char *) A(addr);
            arg += sizeof(unsigned int);
            n++;
            ...
      } while (addr);
      return n - 1;
}

asmlinkage long sys32_execve (char *filename, unsigned int argv,
      unsigned int envp, int dummy3, int dummy4, int dummy5, int dummy6,
      int dummy7, int stack) {
      struct pt_regs *regs = (struct pt_regs *)&stack;
      unsigned long old_map_base, old_task_size, tssd;
      char **av, **ae;
      int na, ne, len;
      long r;

      na = nargs(argv, NULL);
      ...
      ne = nargs(envp, NULL);
      ...
      av = kmalloc( (na+ne+2) * sizeof(*av) , GFP_KERNEL);
      ...
      ae = av + na + 1;
      av[na] = NULL;
      ae[ne] = NULL;
      r = nargs(argv, av);
      ...
      r = nargs(envp, ae);
      ...
```

As seen, the nargs() function is used to count the number of pointer there are in userland. Then a kmalloc() is done, and the nargs() functions is used again to copy the arguments from userland to kernel allocated space. A buffer overflow exists if a concurrent thread changes the pointer array after the count was done, but before the pointers get copied into a kernel allocated buffer.

For uniprocessor machines there is a race condition because both kmalloc() and get_user() can block. On multiprocessor machines kmalloc() or get_user() don't even need to block, because a change to the pointer array can be done in parallel.

Impact:
When properly exploited this yields local root.

Affected versions:
This vulnerability affects all 2.4.x kernels up until 2.4.31. All versions of the 2.6.x kernel are affected up until 2.6.6. Only ia64 and amd64 machines with ia32 compatibility in the kernel are affected.

Suggested Recommendations:
If you are affected by this vulnerability you should update your kernel as soon as possible.

Credits:
Ilja van Sprundel found this vulnerability.

About us:
Suresec Ltd is a global service provider of Internet security solutions and consultancy with unmatched quality from our world class consultancy practice.

Our consultants have pioneered in the field of security research and have closely worked with leading software companies and service providers to mitigate risks and fix a number of critical vulnerabilities, suresec also works closely with a number of open source companies to provide them with a source code auditing and technical consultancy. We have a strong team consultants spread across Europe, the United States of America And Australia specializing in security consulting.