Linux Kernel Programming Eğitimi

Süre: **5 Gün**

BİLGİNÇ
IT Academy
info@bilginc.com
bilginc.com

Bu beş günlük Linux güvenlik eğitimi, Linux çekirdeğinin mimarisini ve çalışma mantığını detaylı şekilde ele alır. Katılımcılar, çekirdeğin temel alt sistemlerini inceleyerek modern Linux kernel kaynak kodu üzerinde uygulamalı çalışmalar yapar.

Eğitim boyunca katılımcılar hem teorik bilgi edinir hem de pratik çalışmalar gerçekleştirir. Özellikle örnek bir **character device driver** geliştirme ve genişletme süreci üzerinden Linux çekirdeği ile doğrudan çalışma deneyimi kazanılır.

Linux Kernel'in farklı sürümler boyunca nasıl geliştiği incelenir ve **Linux Kernel 6.17 sürümüne kadar gelen mimari değişiklikler** ele alınır. Ayrıca kullanıcı modu ile kernel modu arasındaki farklar, süreç planlama mekanizmaları, bellek yönetimi, donanım aygıtlarının yönetimi ve kernel güvenlik mekanizmaları detaylı şekilde açıklanır.

Eğitim boyunca gerçek kernel kodu analiz edilir, yürütme yolları takip edilir ve güvenli kernel geliştirme için önerilen en iyi yöntemler uygulanır.

Programın sonunda katılımcılar:

- Linux Kernel kaynak kodlarını analiz edebilir
- Device driver geliştirebilir
- Kernel davranışını güvenlik ve performans açısından yorumlayabilir

Önkoşullar

Bu eğitime katılacak kişilerin aşağıdaki konularda temel bilgi sahibi olması önerilir:

- User-mode programlama kavramları
- POSIX standartları ve sistem çağrıları
- C dili ile sistem programlama deneyimi

Kimler Katılmalı

Bu eğitim özellikle aşağıdaki profesyoneller için tasarlanmıştır:

- Linux tabanlı sistemler için **device driver geliştiricileri**
- **Malware, rootkit ve kernel seviyesinde tehdit analizi** yapan siber güvenlik uzmanları
- Linux Kernel mimarisini öğrenmek isteyen yazılım mühendisleri
- Düşük seviyeli sistem entegrasyonu veya performans optimizasyonu yapan teknik uzmanlar

Linux Kernel Programming Eğitimi

Süre: **5 Gün**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Neler Öğreneceksiniz

Eğitim sonunda katılımcılar:

- Linux Kernel mimarisini ve alt sistemlerini açıklayabilir
- Kernel mode ve user mode arasındaki farkları anlayabilir
- Linux Kernel kaynak kodunu analiz edebilir
- Çalışan bir Linux character device driver geliştirebilir
- Süreç planlama ve thread yönetimi mekanizmalarını kavrayabilir
- Kernel içindeki sanal bellek yapısını anlayabilir
- Sistem çağrılarını izleyebilir ve analiz edebilir
- Kernel debug ve tracing araçlarını kullanabilir
- task_struct gibi çekirdek veri yapılarıyla çalışabilir
- Güvenli kernel modül geliştirme tekniklerini uygulayabilir
- Linux güvenlik mekanizmalarını (capabilities ve security modules) açıklayabilir

Eğitim İçeriği

Linux Kernel'e Giriş

- Linux mimarisi ve tasarım yaklaşımı
- Kernel sürümlerinin gelişimi (2.6 → 6.17)
- Kernel modu ve kullanıcı modu
- Linux'ta aygıt yönetimi ve donanım soyutlama
- Görev planlayıcı (scheduler) yapısı
- I/O planlama mantığı

Kernel Debug ve İzleme

- Kernel hata ayıklama teknikleri
- strace ve ltrace ile sistem çağrısı analizi
- tracefs ile kernel izleme
- dmesg ve /proc/kmsg logları

Linux Kernel Programming Eğitimi

Süre: **5 Gün**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

- /proc ve /sys sanal dosya sistemleri

Kernel Kaynak Kodlarıyla Çalışma

- Linux Kernel kaynak ağacı yapısı
- Alt sistemlerin klasör yapısı
- Linux Cross Reference ile kod analizi
- Kernel derleme ve cross compile işlemleri

Linux Kernel Boot Süreci

- Firmware'den kullanıcı alanına kadar boot süreci
- BIOS ve EFI boot mekanizmaları
- Boot loader yapısı
- Kernel başlangıç fonksiyonları
- Intel ve ARM platformlarına özgü farklar

Kernel Programlama Rehberi

- Kernel geliştirme için en iyi uygulamalar
- Taşınabilir kernel modülleri yazma
- Kernel veri yapıları ve linked list
- Senkronizasyon mekanizmaları
- Spinlock, semaphore ve atomic işlemler
- printk, /proc ve seq_file kullanımı

Kernel Alt Sistemleri

- Sistem çağrısı mekanizmaları
- Interrupt yapıları
- Process ve thread yönetimi
- Scheduling politikaları

Device Driver Geliştirme

- Character device driver temelleri
- Major ve minor numaraları

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Aygıt kayıt işlemleri
- udev ve hotplug entegrasyonu
- sysfs arayüzleri

Gelişmiş Kernel Konuları

- Kernel zaman yönetimi
- Timer mekanizmaları
- Virtual memory mimarisi
- kmalloc ve kfree ile bellek yönetimi
- Slab ve buddy allocator yapısı

Interrupt Yönetimi

- Kernel modunda interrupt handling
- IRQ yönetimi
- Interrupt handler yazımı
- Bottom halves, tasklet ve work queue yapıları

Linux Ağ Mimarisi

- Linux networking stack
- Network device yapıları
- Ethernet temelleri
- Netfilter ve paket filtreleme

Linux Kernel Güvenliği

- Linux capabilities modeli
- Kernel güvenlik riskleri
- Linux Security Modules (LSM)
- SELinux genel bakış