**23C. Vysvětlete co znamená kernel v OS Linux a pohovořte o historickém vývoji. Z jakých částí se kernel skládá, jak je zabezpečen přístup vyšších vrstev OS k jádru (SYSCAL).**

Kernel v operačním systému Linux je základní komponenta, která spravuje hardware a poskytuje prostředí pro běh aplikací. Jedná se o monolitické jádro, které zprostředkovává komunikaci mezi hardwarem (procesor, paměť, disky) a softwarem (aplikace, služby). Kernel řídí procesy, paměť, vstup/výstup a zajišťuje bezpečnost a stabilitu systému. V Linuxu je kernel open-source, což znamená, že jeho zdrojový kód je volně dostupný a komunitou neustále vylepšován, díky čemuž je Linux flexibilní a široce používaný od serverů po mobilní zařízení.

Historický vývoj Linuxového kernelu začal v roce 1991, kdy Linus Torvalds, tehdy student, vytvořil první verzi jako alternativu k tehdejším systémům, jako byl Minix. První kernel (verze 0.01) byl jednoduchý, podporoval jen základní funkce pro procesory Intel 386. Díky otevřenému modelu vývoje přispěly tisíce vývojářů z celého světa, a kernel se rychle rozrůstal. Klíčové milníky zahrnují verzi 2.6 (2003), která přinesla lepší podporu vícejádrových procesorů, a verzi 5.0 (2019), zaměřenou na moderní hardware a bezpečnost. Dnes je kernel spravován přes systém verzování (např. 6.x) a stále se vyvíjí pod dohledem Linux Foundation.

Linuxový kernel se skládá z několika hlavních částí: **správce procesů** (řídí plánování a běh procesů), **správce paměti** (alokuje a uvolňuje paměť), **systém vstupu/výstupu** (komunikuje s periferiemi, např. disky, síťovými kartami), **síťový subsystém** (zajišťuje síťovou komunikaci) a **ovladače zařízení** (umožňují komunikaci s konkrétním hardwarem). Důležitou součástí jsou také **systémové volání (syscall)**, což je rozhraní, přes které aplikace žádají kernel o služby, například čtení souboru nebo odeslání dat. Tato modularita umožňuje přizpůsobit kernel různým zařízením, od IoT po superpočítače.

Zabezpečení přístupu vyšších vrstev OS k jádru je zajištěno mechanismem **syscall** (systémových volání). Aplikace běží v uživatelském režimu (user space) s omezenými právy, zatímco kernel běží v privilegovaném režimu (kernel space). Když aplikace potřebuje přístup k hardwaru nebo systémovým zdrojům (např. otevřít soubor), odešle syscall, což je standardizovaný požadavek zpracovaný kernelem. Kernel ověří oprávnění a provede operaci, čímž chrání systém před neoprávněným přístupem. Mechanismy jako SELinux nebo AppArmor dále posilují bezpečnost omezováním akcí, které může aplikace provést, a chrání kernel před zneužitím.