

KasperskyOS<sup>®</sup>

# KasperskyOS

Security experts agree that strict enforcement of proper security policies plays the key role in securing the system. But if the issue is just about policy, why are systems still insecure?

## Introduction

Many modern computer systems – including systems that form part of our critical infrastructure, the Internet of things and machine-to-machine communication – run the daily gauntlet of numerous and varied cyber threats. A lot of these systems have specific security objectives related to their features and special aspects of their use.

To achieve these objectives the system needs to implement an appropriate security policy – and that policy needs to be firmly enforced. Indeed, without that enforcement, security becomes a drain on resources as it attempts to plug multiple holes. Security experts agree that strict enforcement of proper security policies plays the key role in securing the system. But if the issue is just about policy, why are systems still insecure?

Common-purpose operating systems are incapable of conforming to the precise security policies of each critical application because common-purpose solutions are flexible and versatile but not intended to be secure by nature.

A special-purpose system is likely to implement its specified policy with some guarantees. But it is both difficult and expensive to assure full implementation of each specific set of security requirements – and the process needs to begin from the ground up in each individual system.

There is a gap in the market for products that implement diverse security policies for systems requiring security assurance. KasperskyOS aims to close this gap by providing a high-assurance secure platform which is capable of enforcing any given policy for different critical applications.

## Purpose

KasperskyOS aims to protect software and data systems from the consequences of:

- the intrusion of malicious code
- viruses
- hacker attacks

These can provoke harmful behavior in any part of the system, potentially resulting in:

- loss or leakage of sensitive data
- reduced performance
- denial of service

In addition, KasperskyOS reduces the risk of harm caused by:

- program bugs
- unintentional mistakes
- premeditated malicious activity

## Implementation

- Enterprise systems
- Special-purpose computer systems
- The Internet of things
- Smart grids
- Industrial systems
- Telecommunications equipment
- Transportation systems
- Critical infrastructure

## Technical requirements

- POSIX API (~98% API) compatible
- Intel x86, x64 и ARM (v6, v7, v8)

## Patents

The technologies that form the basis of KasperskyOS and Kaspersky Security System are covered by a set of patents:

US 7386885 B1, US 7730535 B1,  
US 8370918 B1, EP 2575318 A1,  
US 8522008 B2, US 20130333018 A1,  
US 8381282 B1, EP 2575317 A1,  
US 8370922 B1, EP 2575319 A1,  
US 9015797 B1, DE 202014104595 U1.

# Features

**Unified low-level interprocess communication.** The microkernel of the operating system implements low-level IPC as the only way for processes to interact.

**Complete mediation in access control.** Interprocess communications are controlled according to the desired security policy and this control cannot be bypassed by any means.

**Communications by type.** Communications are categorized by type and properly handled by security runtime to fill the semantic gap between low-level IPC and high-level security policies.

**No anonymous messaging.** All processes and their types of permitted communication must be configured before execution; any other interaction is denied by default.

**Separated security server.** The enforcement of a specific policy is not a security kernel concern but is the sole task of an independent engine that returns a verdict on whether access is permitted or not.

**Security domain separation.** KasperskyOS efficiently separates security domains – confined groups of applications with a restricted influence on each other. This does not preclude the possibility of interdomain communication, if explicitly allowed.

**Diverse policy enforcement.** An independent security engine can enforce the policy that best matches the identified security objectives. The security policy can also be individually configured for every application in the system.

**Tamperproof configuration and services.** The security configuration is stored in protected memory and can be accessed by trusted services only, which are also restricted in their communications.

# Advantages

**Proprietary microkernel and independent security engine.** KasperskyOS is based on a reliable microkernel that implements the only way of communicating. This lightweight microkernel can be used on various platforms. At the same time, the loosely coupled security engine makes it possible to replace the in-house microkernel with another kernel if necessary.

**Multi-level compatibility.** While the system is kept mostly POSIX compatible, the use of a native API further guarantees the secure behavior of applications. The developer can choose how to keep a proper balance between program code compatibility and security.

**Mandatory identification and labeling.** All applications in KasperskyOS are accompanied by their security configuration. Nobody can install an application without installing its relevant behavior configuration. Hardware and application-level resources (files, databases, network ports, etc.) are labelled with appropriate security attributes. It is impossible to access a resource that doesn't have a security label.

**Initially secure system.** KasperskyOS is designed with security in mind and remains secure during its whole lifecycle.

**Modular design.** A modular approach to system design minimizes the footprint of the trusted base and makes it possible to build each individual solution on a case-by-case basis.

**Secure architecture of applications.** Application design is based on a component model that makes secure development easy and elegant.

**Easy-to-configure policies.** IPC types and simple configuration language help to easily define the rules of interprocess communication and access control.

**Verifiability.** Strict adherence to security concepts in system design and implementation makes it possible to verify the security of all solutions based on KasperskyOS.

[www.kaspersky.com](http://www.kaspersky.com)

© 2017 AO Kaspersky Lab. All rights reserved. Registered trademarks and service marks are the property of their respective owners.

Find out more at [os.kaspersky.com](http://os.kaspersky.com)  
All about Internet security: [www.securelist.com](http://www.securelist.com)