Updated 2026-03-13

AbittiCandidate Application for Windows - Documentation for School IT Administration

Quick Reference for IT admins
Quick Reference: Installation on school-managed devices (central IT installation)

Table of Contents

This document provides a comprehensive guide to the Abitti Candidate application for Windows (AbittiCandidate) used in general upper secondary schools for the Matriculation Examination. It is created by the Matriculation Examination Board (MEB), Finland.

The document is intended for school IT administrators and aims to address common questions about the operation of the AbittiCandidate application: purpose, installation, the separate Abitti user account, password settings, information security, technical monitoring, and other administrative aspects. The document is organised into structured chapters and includes tables and highlighted text boxes to aid quick reference.

The purpose of AbittiCandidate application

Abitti 2 works in a client-server -model. The server provides an HTML5 desktop with the test and applications which the candidate can use during the test. The client (AbittiCandidate):

- locks the candidate into the browser so that they can only access the test and it's materials
- monitors activity on the candidate’s workstation

In other words the candidate is "locked" to the kiosk browser. Any malicious activity that cannot be blocked is logged on the server.

How the application works

The candidate uses the AbittiCandidate application as follows:

1. The candidate logs out from their own Windows user account.
2. The candidate logs in to local user "Abitti", which starts the AbittiCandidate application and the technical monitoring. This account is created by the AbittiCandidate installer.
3. When the candidate has participated to the test, they exit from the AbittiCandidate application.
4. When the AbittiCandidate application exits for any reason, the operating system logs out of the Abitti user session.

Why the local Abitti account

Instead of running the exam system on the user’s regular account the AbittiCandidate installer creates a local account to run the exam system. Here are some reasons why:

From the candidate’s perspective:

- **Automatic test launch**: Immediately upon login, AbittiCandidate launches in full screen automatically. The candidate does not see the standard Windows desktop, taskbar, or other applications – only the test application window.
- **Locked interface**: On the Abitti account, all other uses of the device are blocked. No other applications can be opened, menus are unavailable, and tools like Task Manager or File Explorer cannot be accessed. Files belonging to other users on the computer are also not visible from the Abitti account. Applications from the candidate's personal account cannot interfere with taking the test. This reduces the risk of accidental rule violations.
- **No changes to personal data**: The AbittiCandidate installation does not need any access to the candidate’s private data or application settings, because the test operates in the anonymous local account.

- **Targeted monitoring**: Technical monitoring is limited to the exam environment only. When the test is conducted on a separate account, monitoring mechanisms observe only the Abitti account's activity and do not intrude on the candidate's normal user profile or private data.

From the perspective of the IT administrator:

- **User rights**: The Abitti account is a standard, restricted local user without administrator privileges. It cannot make system changes, install software, or access other users' data. This minimizes the risk of misuse.
- **The application starts automatically**: Logging in with the Abitti account does not start a normal Windows session. Instead, the candidate's application opens immediately and all other device usage is blocked. If the application exits with an uncontrollable error, the operating system automatically logs the user out.
- **Other user profiles remain unaffected**: The AbittiCandidate application uses only a separate Abitti account, so installing or running the test does not make any changes to the candidate's normal Windows profile. This protects users' personal settings and files from all exam-related modifications.
- **Disposable settings**: The installer makes several registry and configuration changes to the Abitti user profile to ensure the above restrictions and to enable technical monitoring. These changes are made by the installer so the Abitti account can be removed and recreated by reinstalling the AbittiCandidate application. The Abitti profile does not contain any data that should be preserved.

Logging in to the Abitti account

- **On self-managed (candidate-owned) Windows devices:** The Abitti account appears automatically in the Windows login screen once the AbittiCandidate application has been installed and the computer restarted. The candidate selects the Abitti account from the login menu.
- **On school-managed devices** (e.g., Entra joined or AD domain computers): The Abitti account may not be visible in the menu by default. In such cases, login is done by selecting *"Other user"* and entering the username as `.\Abitti` (dot + backslash + Abitti). This tells Windows to log in with the local account named Abitti.

After the test

When the test ends, the candidate closes the test application, which automatically logs the user out of the Abitti account. The computer returns to the normal login screen, where the candidate can log back to their personal account and continue normal device use. Any actions or trial recordings during the Abitti session are not stored in the candidate's own profile and remain within the Abitti profile.

Installation and administration

The installation process is straightforward:

1. Download the latest AbittiCandidate application from the Abitti support website abitti.fi.
2. The MSI installation package (`AbittiCandidateInstaller.msi`) and the executable binaries are digitally signed by
   Certificate subject: `Ylioppilastutkintolautakunta` Issuer: `GlobalSign GCC R45 EV CodeSigning CA 2020, GlobalSign nv-sa, BE` .
3. Install the package. If an existing installation is found, it will be automatically removed prior to installation.

4. The installation process removes existing Abitti user account and creates a new one. This also resets the password of the Abitti account.

Installing AbittiCandidate application

The AbittiCandidate comes as an MSI installation package named `AbittiCandidateInstaller.msi`. It can be installed in two main ways:

- By the candidate on their own device (e.g., personal laptop).
- Centrally by IT administration on managed devices (such as school-owned computers managed with Intune/ConfigMgr/other solution).

Below, both scenarios are described, along with update-specific considerations.

Installation on a candidate's own (self-managed) device

On a candidate's personal Windows 11 computer, installing AbittiCandidate follows theses steps:

1. Download the MSI installation package.
2. Verify that the installation package is digitally signed by `Ylioppilastutkintolautakunta`.
3. Double-click the installation file to start the installation.
4. Accept the request for administrative privileges.

Updating follows the same process as the initial installation: by downloading the new `AbittiCandidateInstaller.msi` package and running it.

The installation is designed so that the new version automatically removes the old version and the Abitti user account, then installs itself and recreates the Abitti account.

During the first installation, the computer may need to be restarted so that the Abitti account appears on the Windows login screen.

Note: If the device is managed by the school's IT system (e.g., Entra ID, Intune), even if it is otherwise self-managed by the candidate, a blank default password may be disallowed.
In this case, the candidate must follow the school's instructions (for example, using an installation command with a password parameter). See below.

Quick Reference: Installation on school-managed devices (central IT installation)

- AbittiCandidateInstaller is an MSI installer. Updates are done by installing the new version over the old one (no uninstall needed).
- The installer creates a local user account named `Abitti`.
  - On domain-joined devices, login is typically done using: `.\Abitti`
- Managed devices often require setting a password for the Abitti user during installation using the parameter:
  `ABITTI_USER_PASSWORD=YourPassword`

Install command

```
msiexec /i AbittiCandidateInstaller.msi /qn /l*v C:\Windows\Temp\Install-
AbittiCandidate.log ABITTI_USER_PASSWORD=StudentExam2
```

Uninstall command

(Replace with version-specific Product Code, e.g., `{8C4BC1FD-C3BE-453B-996C-84DA1FE109AF}`):

```
msiexec /x "{PRODUCT_CODE}" /qn /l*v C:\Windows\Temp\Uninstall-AbittiCandidate.log
```

Detection rule

- Type: `MSI Type`
  Product Code: `{PRODUCT_CODE}` (version-specific)

Supersedence / Update

- New versions can be installed directly over old ones.

Installation on school-managed devices (central IT installation)

For school-maintained Windows 11 computers managed through systems such as Entra ID, Intune or a traditional AD domain, the AbittiCandidate application is typically installed centrally as follows:

- **MSI package deployment**: Since AbittiCandidate is provided in MSI format, it can be installed via Microsoft Intune (as a Win32 application), Configuration Manager (ConfigMgr) or any other compatible device management solution.
  In Intune, the `.msi` file can be wrapped into an `.intunewin` package; in ConfigMgr and most other management tools, the `.msi` is typically used directly.
- **Installation parameters**: In many managed environments, a password is set for the Abitti account during installation because blank passwords may be disallowed by local policy or compliance settings. This is done by adding the parameter `ABITTI_USER_PASSWORD="Password123"` to the installation command.
  - **Example command (Intune/ConfigMgr or similar):**
    `msiexec /i AbittiCandidateInstaller.msi /qn ABITTI_USER_PASSWORD=StudentExam2`
  - `/qn` = silent installation; `/l*v` can be added for logging if needed.
- **Silent installation with logging and password setting**:
  `msiexec /i AbittiCandidateInstaller.msi /qn /l*v C:\Windows\Temp\Install-AbittiCandidate.log ABITTI_USER_PASSWORD=StudentExam2`
- **Note**: The password may appear in plain text in the log. Storing logs in `C:\Windows\Temp` prevents standard users from reading them. However, for example, Intune Management Extension logs installation commands in plain text and may be readable by a user.

Intune and ConfigMgr can read default commands from the MSI, but the password parameter must be added manually.

When the installation is deployed from the management tool, it installs AbittiCandidate, creates the local Abitti account, sets the given password, and configures settings in the Abitti user profile. Installation is quick – usually only a few minutes – after which the device is ready for Abitti 2 use.

In managed workstations, logging into the Abitti account usually requires entering the username field as: `.\Abitti`

Detection rule for AbittiCandidate

MSI Product Code can be used as detection rule for example with Intune or ConfigMgr. Product Code is unique in each AbittiCandidate version.

Rule type: `MSI`
MSI product code: `usually fills in automatically, example value: {8C4BC1FD-C3BE-453B-996C-84DA1FE109AF}`

management system should fill MSI product code value automatically.

Alternative detection method with .exe file

Rule type: `File`
Path: `C:\Program Files\Abitti`
File: `AbittiCandidate.exe`
Detection method: `String (version)`
Operator: `Equals`
Value: `1.11.1` Example value, change to current version

Updating AbittiCandidate application

A new version can be deployed directly over the old version.

As with self-managed devices, installing the MSI automatically removes the old AbittiCandidate version and the Abitti account, then recreates them.

In Intune, the *Supersedence* function can be used to indicate that the new package replaces the old one. In ConfigMgr, the new application version can be marked as replacing the previous one.

Important: Always include the `ABITTI_USER_PASSWORD` parameter in new deployments, as the update process resets the password.

Uninstalling AbittiCandidate application

If the application needs to be removed from a device, the MSI uninstall command can be used.

Each version has a different `ProductCode`, so the correct uninstall command must be retrieved from that version's details (Intune/ConfigMgr usually handles this automatically).

Uninstalling the application also removes the Abitti user profile, so the device will not be test-ready until AbittiCandidate is reinstalled.

Example uninstall command (replace GUID with actual ProductCode):
`msiexec /x "{8C4BC1FD-C3BE-453B-996C-84DA1FE109AF}" /qn`

Uninstall with logging:
`msiexec /x "{8C4BC1FD-C3BE-453B-996C-84DA1FE109AF}" /qn /l*v C:\Windows\Temp\Uninstall-AbittiCandidate.log`

Troubleshooting installation and runtime issues

If the installation fails or the Abitti account does not appear as expected, check the following:

- **Has the computer been restarted after installation?**
  On Windows Home devices, a restart is required during the first installation for the new user to appear in the login menu.
- **Are there error messages in the log files?**
  The most common cause of failure in managed installations is a missing or non-compliant

password with the `ABITTI_USER_PASSWORD` parameter.

- **AbittiCandidate log locations**

  AbittiCandidate creates several log files that can help diagnose installation or runtime issues.

  Application troubleshooting logs

  - `C:\ProgramData\Abitti\Logs`

  This folder contains logs generated by AbittiCandidate components and is the primary location to check when troubleshooting application startup or runtime issues.
  These logs are particularly useful if the AbittiCandidate application fails to start or shows a blank screen (for example, if the application crashes during startup).

  Installation-related logs The installation process also creates logs in the `C:\Windows\Temp` directory:

  - `AbittiUserRemoval-yyyymmddxx.log`
  - `AbittiUserCreation-yyyymmddxxx.log`
  - `AbittiTpmCheck-yyyymmddxxx.log`

  If `/l*v` is used during installation, an MSI log file can also be created, for example:

  - `C:\Windows\Temp\Install-AbittiCandidate.log`

If issues occur, reinstalling is usually the simplest solution: run the same installation package again to repair. This restores missing components (remember to include the password parameter if needed).

If necessary, uninstall and reinstall the application from scratch. Remember, the Abitti profile contains no data or settings to be preserved.

Extreme cases: Sometimes a process gets stuck and prevents the Abitti profile from being removed. This could happen, for example, if an antivirus program locks a file in the Abitti user's profile folder. In such situations, follow the provided instructions from MEB Support to clean up the Abitti user account manually or remotely through the management system.

Collect Troubleshooting Report Automatically (AbittiCandidate Crash or Install Issues)

If AbittiCandidate crashes (for example, showing only a black screen) or fails to install/update, you can run a built-in SupportTools script to gather a troubleshooting report. This report (logs and system info) will help MEB Support diagnose the issue.

- **Navigate to the SupportTools folder:**
  Open `C:\Program Files\Abitti\SupportTools\`
- **Run the log-gathering tool:**
  Right-click
  `Gather_AbittiCandidate_logs - RUN ME (possibly with Run as Administrator).bat`
  and preferably select option **Run as Administrator**.

  > ***Note:*** *While not required, running the tool with admin rights collects more information (and is needed to capture installation log files when troubleshooting install issues).*

- If possible **Select Extended Report:**
  When prompted in the tool's window, choose option `2` for an extended report (this is the

default and collects more information).

- **(Optional) Provide a summary:**
  You can also provide a short description of the problem.

- **Find the output folder:**
  After the tool finishes, it creates a folder named `AbittiCandidate_Troubleshooting` on the Desktop of the user who ran the tool.
  (If you ran it using "Run as Administrator" with a different account, check that account's Desktop.)

- **Locate the report files:**
  Inside the troubleshooting folder (in the subfolder named with the current date/time), you will find:

  - an **HTML** report file (overview of the issue)
  - a **ZIP** file containing detailed logs

- **Send files to support:**
  Send **both** the HTML and ZIP files to MEB Support for analysis.

Abitti user account password options

By default, the Abitti user account is created without a password when the candidate installs the application themselves.
In most cases for centrally managed devices, a password needs to be set during the installation command (remote management).

This section covers:

- What the default password is in different situations.
- How the password can be set during installation.
- How it can be changed afterwards.
- What happens to the password during updates.
- How password policies (e.g., complexity requirements) affect the Abitti account.

Default password during installation

- **Candidate self-managed device:** By default, the Abitti account is created without a password, meaning the password field can be left blank when logging in. This is intentional, to allow easy login without risk of forgotten passwords.
- **Centrally managed device** (e.g., Active Directory/EntraID/Intune/ConfigMgr/other managed environment): In many managed environments, a blank password may be disallowed by organizational policy, compliance settings, or local security requirements, so the Abitti account must be given a desired password as part of the installation command. The chosen password must meet Windows' basic requirements (appropriate length, mix of letters and numbers).

Setting the password during installation (by IT administration)

In most cases, IT administration needs to set a password for the Abitti account during installation.
This is done by adding the `ABITTI_USER_PASSWORD` parameter to the installation command. Example:

- **Intune/ConfigMgr/other management tool installation command:**
  `msiexec /i AbittiCandidateInstaller.msi /qn ABITTI_USER_PASSWORD=YourDesiredPassword1!`

(Replace `YourDesiredPassword1!` with the desired string.)

- For a candidate's self-performed installation, this installation command is rarely needed. An exception is when the candidate has an IT-managed device but has admin rights and installs the AbittiCandidate application themselves. In such cases, domain policies will likely block a blank password, and the candidate must install the application using the above command.

Typically, the password parameter is intended for management tools.
The administrator-set password can be something agreed upon for the entire school (if you want everyone to have the same password) or even unique per device.

MEB does not require a specific password, nor does it mandate whether it must be blank or complex. This decision is up to the school's own test and information security policy.

Changing the password afterwards

After deployment, there are several ways to manage the password:

- **Candidate can set/change the password** on their own device if desired. In Windows account management, the Abitti account appears and a password can be added (if it was blank) or changed. This may be relevant if the candidate wants to prevent others from taking the test on their device. In such cases, the candidate is responsible for remembering or changing the password as needed before the test.
- **IT administration can centrally change the password** when necessary. For example, a script or other deployment tool can be used to update the Abitti account password remotely on all devices. This could be done before a test day: set a new common password on all devices and inform the invigilators/candidates.
- **Self-service portal option**: In some environments, a self-service mechanism can be offered through an application portal, allowing, for example, a teacher or candidate to reset the Abitti password without reinstalling the application.

In practice, many schools choose to keep the Abitti password known in advance and fairly simple (or blank on self-managed devices) for simplicity.

Example: Managed devices may have the Abitti password set during installation to `StudentExam2` (complex enough to meet requirements). This password is then always used for login unless otherwise decided.

Password reset during updates

A very important note: **Every time the AbittiCandidate software is updated, the Abitti account password resets to the default.**
The installer always creates a new Abitti user (after deleting the old one), and it cannot know what the previous password was. Therefore:

- If the device is candidate-owned, after the update, the new account again has no password.
- If the device is managed and IT set a password, the update installation will set the password to whatever is specified in the new installation package parameters.

Conclusion: Always include your desired `ABITTI_USER_PASSWORD` parameter in the installation command when updating AbittiCandidate to keep the password as intended.

Security notes regarding passwords

- If the installation is done with a password in the parameter, it will be recorded in plain text in some log files.
  For example, the Windows Installer log (if `/l*v` is used) contains the command parameters, and if installed via Intune, the corresponding log and password parameter may be readable by a standard user.

  Recommendation: Restrict read permissions for installation and uninstall logs. For example, save logs in `C:\Windows\Temp`, which is not readable by normal users. Alternatively, use a method where the password is not passed directly on the command line, e.g., run the installation via a separate script that hides the password.

- **Blank password behaviour in Windows:**
  A blank password is a special case in Windows. By default, Windows applies the policy **"Accounts: Limit local account use of blank passwords to console logon only."**
  This means that a local account without a password can normally only be used at the physical Windows sign-in screen.

  In practice this prevents many credential-based authentication methods, including tools such as `runas`, because Windows does not allow blank passwords to be used for these logon mechanisms.

  For this reason, in the specific case of the Abitti account, leaving the password blank may actually reduce certain misuse scenarios compared with a password-protected account.

- **Password complexity:** While MEB does not require a complex password, your school's security policy might. It's recommended to choose a password with at least 8 characters, a mix of letters and numbers, and preferably special characters.
  Example: `AbittiExam2024!` is memorable but meets requirements. If the complexity policy blocks the word "Abitti," something like `StudentExam2!` could be used.

> ***Important***
>
> *For the Abitti account, a blank password does not necessarily reduce security.*
> *Because Windows restricts blank-password local accounts to console logon only by default, the absence of a password may actually prevent certain misuse scenarios that require credential-based authentication methods.*

Summary

Managing the Abitti account password is flexible. You can keep it blank on candidate devices, set consistently on managed devices, or change it as needed.

The most important thing to remember is that the security characteristics of the Abitti account depend on whether the account has a password or not.

A blank password benefits from Windows' built-in restriction that limits such accounts to console logon only. If a password is configured, Windows treats the account more like a normal local user account, which means the credentials may potentially be used by other mechanisms such as `runas` or other credential-based process launch methods.

If organisational policy allows it, leaving the Abitti account without a password is often both the simplest and the technically safest option.

Security considerations

Typically a local user account with a (possibly known) password or even no password at all is not among the best practices when administering Windows workstation. However, in the case of the Abitti account, the risks are significantly reduced due to several built-in safeguards:

- **Physical access requirement**: In practice, the intended way to use the Abitti account is to sit physically at the device. By default, Windows restricts blank-password local accounts to console logon only, which means a blank-password Abitti account cannot normally be used through common remote or alternate credential logon methods.

  If the Abitti account has a password, this built-in restriction no longer applies. In that case, whether the account can be used remotely depends on the organization's other controls, such as firewall rules, enabled remote management services, and local security policies. In many school environments, these controls still reduce remote access, but this should not be assumed automatically.

- **Restricted rights**: The Abitti account is a normal standard user without administrative privileges. It does not belong to the Administrators group, so even if someone logs in to the Abitti account, they cannot make system-wide changes or install malware at the system level.

- **Locked test environment**: As described earlier, logging into the Abitti account provides an interface that is completely locked to the test application. A person logged into the Abitti account has no access to standard tools like Command Prompt, PowerShell, File Explorer, or a web browser. Everything visible on the screen during the exam session is controlled by AbittiCandidate, making it very difficult (if not impossible) to perform any harmful actions because there is simply no freedom to operate.

- **No local data**: When executed, the AbittiCandidate does not read or execute any data/scripts which could be manipulated by malicious attacker who has access to the Abitti local account.

- **No access to other data**: Windows isolates user profiles from each other. The Abitti user cannot see files belonging to other users (such as a teacher's or a candidate's own profile documents). Network drives connected under another user's credentials are also not available in the Abitti profile.

- **Logs and traceability**: Due to technical monitoring, every action performed on the Abitti account during the test can be reviewed afterward. If someone were to find a way to misuse the Abitti environment, traces would likely remain in log files, which could be analyzed and used as evidence.

Password known by everyone

If all candidates are told a common password like `Abitti2024!` for managed devices, could a candidate misuse that information?

In theory, they could start any school computer and log in to the Abitti account. The result would simply be the Abitti candidate application start screen. While the AbittiCandidate is basically a kiosk browser, it connects only to the Abitti 2 server. It cannot be used to browse or execute HTML5 content served by a regular web server.

Maintenance and security

From a general Windows security perspective, it is common best practice to require passwords for all user accounts. However, the Abitti account is a special-purpose local account used only for the exam environment.

Because Windows restricts blank-password local accounts to console sign-in only by default, the absence of a password may actually reduce certain impersonation scenarios. If a password is configured, the account behaves more like a standard local account and the credentials may be usable through other mechanisms such as `runas`.

For this reason, if organisational policies allow it, leaving the Abitti account without a password is often the simplest and technically safest configuration.

Technical enforcement of this is limited, so it would mostly rely on guidance.

In most cases, a blank or simple password for the Abitti account will not cause a security incident as long as other protections (physical invigilation, network settings) are in place.

Comparison to other solutions

Some might wonder whether the Abitti account could have been created in a more “hidden” way, embedded deep in the system.
In fact, transparency (a clearly named account called Abitti) is an advantage here: it is clear exactly what environment is in use.

The Abitti account is part of the official test system for the matriculation examination, and it has not been found to cause harm to other parts of the IT infrastructure.

Summary

When the AbittiCandidate application is installed according to instructions and IT administration has ensured the basic settings (e.g., password policy considerations), the existence of the separate Abitti account does not weaken the security of the device or the network.

On the contrary, it improves overall security by isolating the test environment.

As long as your school is aware of the Abitti account’s use and management, you can be confident in its safety.

Some considerations for IT administration (Intune policies, logs, antivirus, AppLocker, profile cleanup)

Deploying AbittiCandidate may raise specific questions within the organization’s IT environment. This chapter compiles other key IT administration aspects and gives recommendations for handling them.

Issue / Concern	Notes / Recommendation
Password policies and compliance (Intune/AD)	Intune compliance policy or Group Policies in AD may set password requirements for all user accounts (e.g., minimum length, complexity, or expiration). These requirements may also apply to the local Abitti account. In the specific case of the Abitti account, administrators should be aware that requiring a password may reduce some of the built-in protections provided by Windows. A blank local password is normally limited by Windows to console sign-in only, which prevents many alternate credential

	use cases. **Recommendation:** If your environment allows it, consider creating an exception so that the Abitti account can remain without a password. This may require reviewing Intune compliance settings or password requirements applied to local accounts. If a password must be used, ensure that password expiry, forced password change, or complexity enforcement does not interrupt the exam login process when using manually configured password.
Security of installation logs	As mentioned earlier, if you include the Abitti password in the installation command, it appears in plain text in installation logs. In Intune-managed environments, Win32 application installation logs with the install command are typically stored in: `C:\ProgramData\Microsoft\IntuneManagementExtension\Logs\AppWorkload.log`, which is readable by standard users by default. Internal AbittiCandidate installation logs may also show the password, but the default directory `C:\Windows\Temp` is not readable by normal users. **Recommendation:** Consider using a separate installation script (even a one-line wrapper) where the password is set securely in the script instead of being visible directly in Intune logs.
Antivirus alerts	Abitti2 performs technical monitoring (collects system data in multiple ways), which could resemble malware behavior to antivirus software. Monitoring is performed by signed processes, so they are generally trusted. For example, Windows Defender Antivirus and Microsoft Defender for Endpoint have not blocked the monitoring processes in tests. Other antivirus programs have not yet been tested. So far, antivirus has not required extra actions. **If needed:** Add the AbittiCandidate monitoring components as exclusions in your antivirus. At minimum, exclude the process `AbittiMonitoring.exe` and the directory `C:\Users\Abitti\AppData\Local\Abitti\Monitoring\WorkDir`. In Intune, this can be done in Endpoint Security → Antivirus settings, or with Group Policy (Administrative Templates → Windows Components → Microsoft Defender Antivirus → Exclusions). For other vendors' antivirus, create similar exclusions. Always test on a few devices before a live test to ensure no antivirus pop-ups appear. Note: File names and the working directory name were changed in AbittiCandidate version 1.5.0. Previously, the file names were nsa-policy.exe, nsa-windows-amd64.exe and the directory was C:\Users\Abitti\AppData\Local\Abitti\NSA\WorkDir.
AppLocker / application control	If AppLocker or similar application control is used, note the specifics of Abitti2. The main test application `AbittiCandidate.exe` is located in `C:\Program Files\Abitti`, which is typically already in allowed paths (signed programs under Program Files). Monitoring programs, however, run from the Abitti user's profile folder: `C:\Users\Abitti\AppData\Local\Abitti\Monitoring\WorkDir`. AppLocker often blocks execution from under `C:\Users\`. **Recommendation:** Add an AppLocker rule allowing the Abitti account to run executables from that folder, particularly `AbittiMonitoring.exe`, `osqueryd.exe`, and other monitoring tools. Limit the rule to that path and signed binaries. You can use the certificate from `C:\Program Files\Abitti\AbittiLoader.exe` for the AbittiMonitoring.exe rule. Note: File names and the working directory name were changed in AbittiCandidate version 1.5.0. Previously, the file names

	were nsa-policy.exe, nsa-windows-amd64.exe and the directory was C:\Users\Abitti\AppData\Local\Abitti\NSA\WorkDir. If the Abitti account must have a password, application control (e.g. AppLocker or Windows Defender Application Control) can also be used as an additional mitigation. For example, administrators may restrict the Abitti account so that only the Abitti exam application and its monitoring components are allowed to run, while tools such as `cmd.exe`, `powershell.exe`, `pwsh.exe`, `regedit.exe`, `reg.exe`, `taskmgr.exe`, `mmc.exe`, and, where appropriate, `explorer.exe`, are blocked for that account.
Profile cleanup (SharedPC)	Some schools use Windows Shared PC mode or other mechanisms that automatically remove unused user profiles after a set time (e.g., 30 days). This could delete the Abitti profile before the next test, which is a problem because it contains essential settings for monitoring and the environment. Without these settings, the test likely cannot run, and the fix would be to reinstall AbittiCandidate. **Recommendation:** Do not enable automatic profile deletion on devices with AbittiCandidate installed, or add the Abitti account to the exceptions list so its profile is not deleted. In SharedPC mode, follow Microsoft's guidance for exemptions. Note that the Abitti account SID likely changes with each Abitti update, as the account is removed and recreated during installation. If the profile is accidentally deleted, reinstall AbittiCandidate before the next test to recreate it.
Disable Switch user from Ctrl+Alt+Delete view in Abitti account	By default, AbittiCandidate does not block Fast User Switching via Ctrl+Alt+Del, as this requires a system-wide policy change — a machine-level setting that disables the option for all users on the computer, not just the Abitti account. For this reason, the AbittiCandidate installer does not enforce it by default. This leaves a potential loophole where candidates could switch to another account during an exam. **Abitti2 monitoring has always detected such attempts** and today it also generates a post-exam report indicating whether a user switched accounts. **Recommendation:** Use Active Directory Group Policy or Intune Settings Catalog to hide all entry points for switching users. In AD GPO, enable `Hide entry points for Fast User Switching` under `Computer Configuration → Administrative Templates → System → Logon` (or set the registry value `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System HideFastUserSwitching = 1`). In Intune, enable `Windows Logon → Hide Fast User Switching` in the Settings Catalog. This removes the "Switch User" option from the login screen, Start menu and Ctrl+Alt+Del menu, ensuring candidates remain locked into the Abitti account during the test.

Optional hardening: Secondary Logon service

In some environments, administrators have observed that disabling the Windows **Secondary Logon** service (`seclogon`) prevents many common credential-based process launch methods such as `runas` and some PsExec scenarios.

This can reduce the possibility of starting processes as the Abitti account if the account has a known password.

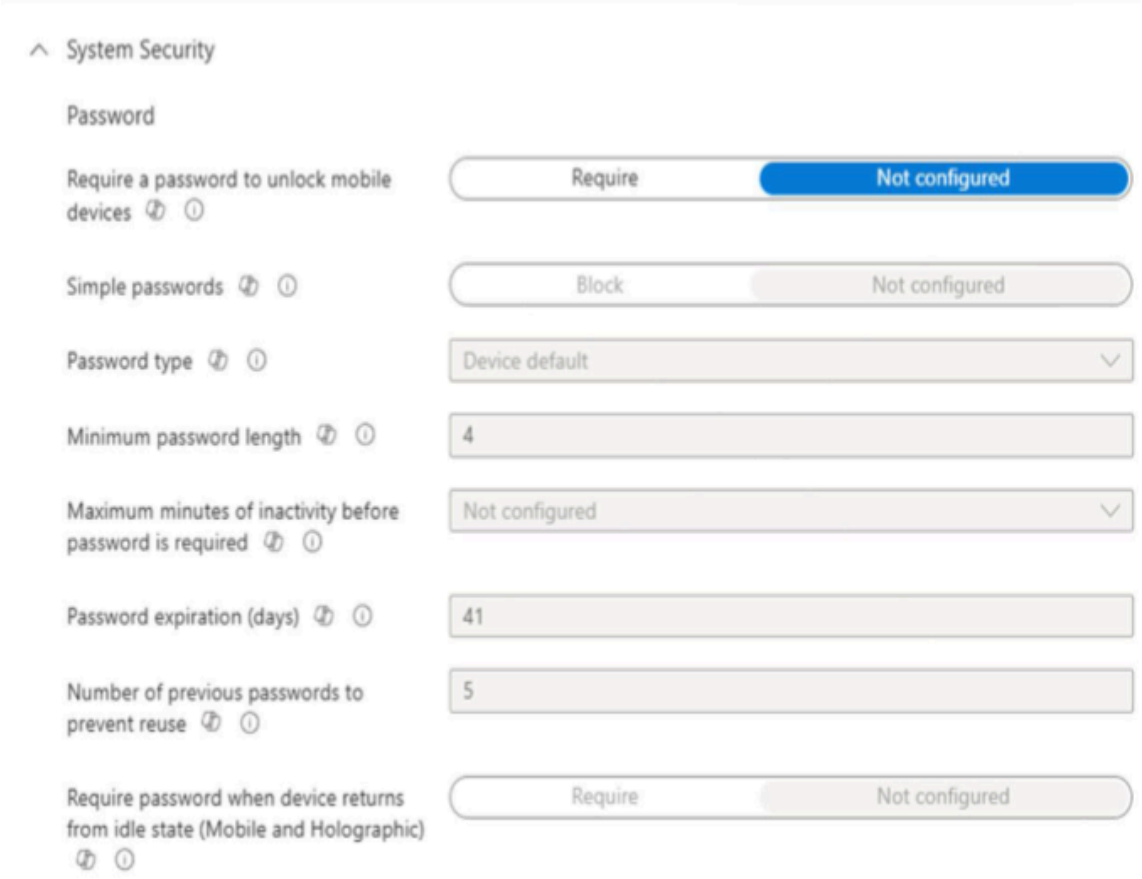
However, this configuration may affect certain administrative workflows or software that relies on "Run as different user" functionality. Because of this, it should be considered an **optional hardening**

measure and tested carefully before deployment. MEB does not require or automatically configure this setting.

Traditional local deny-logon rights (such as denying network, batch, or service logon) may still be useful as general hardening, but they do not by themselves solve the specific problem of starting processes as the Abitti account with known credentials.

To summarise:

- **Compliance policy impact on passwords**: Especially in Intune's Device Compliance Policy settings, there is an option such as "Require Password…" under certain conditions, which in Windows usually refers specifically to the password of local accounts. Simple passwords: Block (i.e., forcing a complex password) can lead to the application installation failing (the user cannot be created with the given "weak" password) or, when logging in with the Abitti account, a forced password change. Password expiry can also result in the Abitti account password being forcibly changed at login. If X number of previous passwords are blocked, then the password must be changed to a unique one that has not been used before.



- If you have Intune (or another) compliance policy in place, ensure that it does not interfere with the candidates' user experience during the Abitti2 test. These should be tested well in advance and over a longer period before the actual tests begin.
- **Antivirus and AppLocker rules**: Abitti2 does not, in itself, store the monitoring data produced on the school computer for long; instead, it is transmitted to the server and deleted

locally. Installation and event logs that remain on the device should be protected as mentioned earlier.

The points presented are not, in principle, actual problems, but it is good to be aware of them in advance. They have been raised from the very beginning of the application's release so that IT administration can plan and test configurations if necessary. By planning the installation and device policies as a whole, you can ensure that everything runs technically smoothly on the test day.

Handling Antivirus Interference and "Vilpinvalvonta ei toimi"

Some antivirus products may mistakenly block AbittiCandidate's technical monitoring tools, causing the error **"Vilpinvalvonta ei toimi"** (technical monitoring not functioning). This topic has become increasingly relevant because monitoring issues now appear in two places visible to schools:

1. **During the exam**, the invigilator's view in the exam server may show "*Valvonta ei toimi*" for an individual candidate, and
2. **After the exam**, an automatic 24-hour report shows two key items:
    - whether the candidate logged in using another Windows user account, and
    - whether monitoring was unavailable during certain periods.

This report is intentionally limited to these two publicly documented items, even though the monitoring system itself performs a significantly broader set of technical checks in the background. Ensuring that monitoring runs reliably — for example by configuring the recommended antivirus exclusions — helps avoid unnecessary warnings in the invigilator view and ensures accurate information in the post-exam report.

This chapter describes MEB's actions to improve compatibility, the issues observed with antivirus products, and the recommended configurations for both school-managed and student-owned devices. Proper allowlisting significantly reduces monitoring failures.

MEB actions to improve compatibility

EV Code Signing
All key AbittiCandidate components — including `AbittiCandidate.exe`, `AbittiLoader.exe`, `AbittiMonitoring.exe`, `AbittiCandidateInstaller.msi`, `CreateUser.ps1`, `RemoveUser.ps1` and `CheckTpm.ps1` — are signed with an Extended Validation (EV) certificate issued to the Matriculation Examination Board. EV signatures significantly reduce the chance of antivirus false positives.
Certificate subject: `Ylioppilastutkintolautakunta`
Issuer: `GlobalSign GCC R45 EV CodeSigning CA 2020, GlobalSign nv-sa, BE`

Microsoft Defender allowlisting
MEB has submitted `AbittiMonitoring.exe` for Microsoft Defender analysis. Current versions of Windows Defender generally trust the monitoring tool, preventing the earlier behaviour where Defender occasionally blocked it. New versions will be re-evaluated as they are released.

Monitoring success tracking
MEB tracks whether monitoring starts successfully during actual tests. Apart from antivirus

interference, no systemic faults have been observed. Almost all "vilpinvalvonta ei toimi" cases originate from antivirus blocking monitoring components.

Observed antivirus interference

Windows Defender (previous issue)

Earlier versions of Windows Defender occasionally blocked the monitoring process. This issue is currently resolved, and keeping Windows fully up to date generally prevents Defender from interfering with technical monitoring. **However, the situation may change with future Defender definition updates.** MEB will continue to submit new AbittiCandidate technical monitoring versions to Microsoft for allowlisting, but until allowlisting is confirmed, there is always a risk that Defender may temporarily block monitoring again. For this reason, **it is still recommended to configure the exclusions listed below on all devices** to ensure stable operation during exams.

Third-party antivirus blocking

Schools have reported interference from **Norton 360**, **Avast/AVG**, **Bitdefender**, and others. These products may:

- Block the AbittiCandidate installer from creating the **Abitti** user account
- Prevent `AbittiMonitoring.exe` from running during the exam
- Remove or quarantine monitoring files from the Abitti profile

This typically causes the candidate's monitoring status to show as **not functioning**, even though nothing is wrong with AbittiCandidate itself.

Recommendations for centrally managed school devices

School IT should configure antivirus settings on all Windows devices used for Abitti2 exams. This prevents last-minute problems during the test session.

Exclude the key AbittiCandidate components

Add exclusions for:

Component	Default Path
AbittiLoader.exe	`C:\Program Files\Abitti\AbittiLoader.exe`
AbittiMonitoring.exe	`C:\Users\Abitti\AppData\Local\Abitti\Monitoring\WorkDir\AbittiMonit`
Monitoring WorkDir	`C:\Users\Abitti\AppData\Local\Abitti\Monitoring\WorkDir\`

AbittiCandidateInstaller.msi	(varies)
MEB code-signing certificate	(Publisher rule)

Allow by publisher (optional, recommended for enterprise environments)

If your antivirus supports publisher-based allow rules, create a rule based on the **EV code-signing certificate**. This automatically trust future updates signed by MEB without manual reconfiguration.

Implement via management tools

- **Intune:** *Endpoint Security → Antivirus → Exclusions* (Process and Path)
- **Group Policy (Defender):** *Computer Configuration → Windows Components → Microsoft Defender Antivirus → Exclusions* (Process and Path)
- **Other AV systems:** Use the vendor's recommended exclusion method

Verify exclusions before exam day.

Recommendations for student-owned devices

Students often use devices with third-party antivirus products. Schools should instruct them to:

- Allowlist:
  - Process: `C:\Program Files\Abitti\AbittiLoader.exe`
  - Process: `C:\Users\Abitti\AppData\Local\Abitti\Monitoring\WorkDir\AbittiMonitoring.exe`
  - Path: `C:\Users\Abitti\AppData\Local\Abitti\Monitoring\WorkDir\`
- Approve or restore any detections of AbittiMonitoring.exe
- If AbittiCandidateInstaller.msi installation is blocked and exclusions cannot be added, student can temporarily disable real-time scanning **only during installation**, offline, and re-enable immediately afterwards (last resort). This method is **not an official recommendation** from MEB due to the risk of forgetting the antivirus off, but some schools have used it successfully.

Students should **not** disable antivirus during the exam. Proper exclusions allow the device to remain secure while enabling monitoring.

General recommendations

- Expect that some antivirus products may interfere without configuration.
- Keep antivirus definitions up to date — vendors often fix false positives.

- **Ensure invigilators know that "Vilpinvalvonta ei toimi" typically indicates antivirus interference, not exam system failure.**
- With correct allowlisting on both school and student devices, antivirus-related monitoring issues can be almost completely eliminated.

Technical monitoring

To ensure fairness in the matriculation tests, AbittiCandidate includes a comprehensive technical monitoring feature. Technical monitoring is a software-based solution that observes events in the test environment and collects evidence of possible cheating attempts.

This chapter explains why monitoring is carried out, what it observes, and how it affects (or does not affect) the candidate’s experience. The details of monitoring are disclosed in the Act on the Openness of Government Activities (621/1999), 24 § paragraph 7.

Why technical monitoring exists

In digital tests, there is always a risk of cheating (e.g., unauthorized use of online resources, programmatic communication with outsiders, etc.). Traditional invigilation (an invigilator walking around the room) is not sufficient to detect all forms of digital cheating. Technical monitoring has been developed to support the test as follows:

- **Preventing** many types of cheating in advance by discouraging attempts when candidates are aware of monitoring.
- **Detecting** suspicious activity in real time, if something obvious happens.
- **Enabling** evidence collection to confirm cheating afterwards using concrete data.

What technical monitoring observes

Without listing every detail, technical monitoring in AbittiCandidate may observe key activities under the Abitti account, such as:

- Screen contents or changes on the screen
- Keyboard events
- Mouse and other peripheral device usage
- Clipboard contents
- Audio device usage
- Network traffic and connections
- Processes and system state

This list is presented as an example only and does not restrict MEB in developing monitoring.

Implementation

When logging into the Abitti account, a monitoring program (or several) runs in the background to collect this information. These programs are optimised so that the quality of the test performance is not affected (they consume very few resources and operate silently in the background).

These programs get the monitoring tasks from the server during the test so the activities may vary during the exams. Furthermore, the activities cannot be analysed by decompiling the monitoring applications.

Before executing the monitoring tasks the digital signature of the tasks is verified to prevent malicious code execution.

How monitoring reacts and appears in different situations

There are three possible operation modes, and AbittiCandidate may apply one or more depending on the situation:

1. **Real-time warning to the candidate**: If something clearly prohibited is detected, the system may display a warning on the screen. For example the candidate may be warned if another user account is still logged in in the background. This can happen if the candidate did not log out of their personal account before logging in to the Abitti account. Such warnings are intended mainly to protect the candidate from unintentional fraudelent behaviour.

2. **Alert to the test invigilator**:
   In certain serious cases, the system can send a notification to the party monitoring the test (e.g., the invigiltors' view). For example, if the monitoring system is not functioning on the candidate's device, an alert may appear in the invigilators' view. The alert may block the candidate from continuing until the reason for the warning has been clarified.

3. **Post-test review**: In most cases, all monitoring data is stored without interfering in real time. After the test, the MEB specialists review the output produced by the monitoring. Furthermore, if a new cheating method is discovered in one test, recordings can be searched afterwards to find all candidates with similar activity.

From the candidate's perspective, technical monitoring is almost unnoticeable. In rare cases, they may receive a warning on screen, as mentioned above.

Impact on information security and privacy

- Technical monitoring runs under the permissions of the local Abitti account. Monitoring programs do not start under other Windows user accounts.
- This means that monitoring is currently limited by the permissions of the Abitti account. In practice, the monitoring can observe anything on the device that is visible or accessible with the privileges of the Abitti account. This includes, for example, information that is not stored inside the Abitti user profile itself but is still visible to that user session.
- The exact scope of monitoring may evolve in future versions, but it will remain limited by the permissions available to the Abitti account unless separately documented otherwise.
- Collected data is sent to MEB via the Abitti 2 exam room server. It is not processed or stored permanently at the school.

Maintaining technical monitoring

From the school IT perspective, it is important to ensure monitoring is not blocked (see the chapter on antivirus/AppLocker).

Otherwise, no special actions are needed – it starts automatically in the Abitti account and closes when logging out.
MEB will provide instructions if antivirus exclusions are needed. So far, no such exclusions have been necessary.

Closing remarks

This document aims to cover all the key aspects of the AbittiCandidate Windows application from the perspective of school IT administration and information security – from installation to monitoring. The MEB is committed to designing and implementing a secure, isolated, and manageable system that contributes to a fair matriculation examination.

By following the instructions given here – especially regarding password practices, security settings and configuration profile management – you should be able to deploy AbittiCandidate without problems and respond to concerns raised by teaching staff or candidates.

If there are still questions or exceptional situations not addressed here, we recommend contacting the Matriculation Examination Board's Abitti support. The contact information can be found at abitti.fi. The website contains updated version of this document and other up-to-date information about AbittiCandidate and Abitti 2.

With thorough planning and trialing, you can ensure smooth and fair matriculation examinations for all candidates.