

---

GabKeyboardHook Активированная  
полная версия Скачать

[Скачать](#)

**GabKeyboardHook Crack+**

Библиотека GabKeyboardHook  
Download With Full Crack обнаруживает  
каждый ввод в текущем целевом  
приложении — даже системный ввод.

С помощью этого ввода вы можете  
выполнять свои собственные  
пользовательские функции, а также  
свои собственные функции, которые  
запускаются горячими клавишами.

Основные особенности

GabKeyboardHook следующие: -  
Обнаруживает входные события  
текущего активного целевого

---

приложения. - Уже включен в стандартную установку Delphi. Никаких дополнительных Delphi DLL или компонентов не требуется. - События ввода обнаруживаются в тот момент, когда пользователь начинает печатать. Таким образом, ввод сохраняется в памяти. Если ввод сработает один раз, то библиотека GabKeyboardHook не потеряет это состояние. - События ввода могут обрабатываться в ваших пользовательских функциях. - Использует только системные вызовы. - Ввод с клавиатуры обрабатывается очень быстро - никаких задержек и проблем с синхронизацией потоков. - Легко показывает, как часто пользователь был вызван. Пример программирования GabKeyboardHook: Если пользователи вашего приложения вводят следующую горячую клавишу: F12 + 1 = клавиша F12 нажата со значением 1 Клавиша F12 нажата со значением 1 ; Если клавиша F12 была нажата со значением 1, ; значение клавиши F12

---

отображается в текстовом поле ; приложения. функция GabInput(const Event: EventRec): String; начинать Ф12-Ф12; конец; Если пользователи вашего приложения вводят следующую горячую клавишу: CTRL+F12 + 4 = Клавиша CTRL+F12 нажата со значением 4 Клавиша CTRL+F12 нажата со значением 4 ; Если клавиша CTRL+F12 была нажата со значением 4, ; значение клавиши CTRL+F12 отображается в текстовом поле ; приложения. функция GabInput(const Event: EventRec): String; начинать CTRL+F12-CTRL+F12; конец; Исходный код GabKeyboardHook предоставляется вам бесплатно.

Пример программирования GabKeyboardHook Вы можете найти исходный код GabKeyboardHook здесь: Github. Поддерживаемые операционные системы: \* Windows - 32 и 64 бит. \* Линукс - 32 и 64 бит. \* MacOS — 32- и 64-битная версии. \*

**GabKeyboardHook Crack+ Full Product Key Free Download [Latest]**

---

1709e42c4c

---

## GabKeyboardHook Keygen For (LifeTime) [32|64bit]

//Описание GabKeyboardHook: //Что  
нового в GabKeyboardHook: // Целевые  
платформы: - Windows 10, Windows 8.1  
- Виндовс 8, Виндовс 7 - Windows XP  
SP3 или более поздняя версия,  
Windows Vista SP2 или более поздняя  
версия //Что нового в  
GabKeyboardHook: //Версия  
//GabKeyboardHook используется с  
версий: 2.3.0 до версии: 2.5.0  
//Используемые библиотеки - Jni.h,  
JniSupport.h, utiljni.h //Используемые  
пространства имен - н/д  
//Используемые функции -  
GabKeyboardHook\_Initialize,  
GabKeyboardHook\_Terminate,  
GabKeyboardHook\_UpdateKey,  
GabKeyboardHook\_GetKeyState,  
GabKeyboardHook\_SetKeyState,  
GabKeyboardHook\_GetModifierState  
//Используемые классы - GCHook  
//Используемые каталоги - н/д  
//Используемые заголовочные файлы -  
н/д //Используемые библиотеки - н/д

---

```
//Используемые пространства имен -
н/д //Используемые функции - н/д
//Используемые классы - н/д
//Используемые библиотеки - н/д
//Используемые каталоги - н/д
//Используемые функции - н/д
//Используемые классы - н/д
//Применение
GabKeyboardHook_Initialize () { вернуть
ИСТИНА; }
GabKeyboardHook_Terminate() {
вернуть ИСТИНА; } int
GabKeyboardHook_SetKeyState (JNIEnv *
env, тип JvmtiEventType, jobject
gkEvent, состояние jint) {
gkEvent->key_pressed = состояние;
вернуть GCHook::UpdateKey(gkEvent,
состояние); } int
GabKeyboardHook_GetKeyState (JNIEnv *
env, jboolean bEnableEvent) { вернуть
GCHook::
```

### **What's New In?**

=====  
GabKeyboardHook  
является частью фреймворка  
GabBinding. GabKeyboardHook состоит

---

из двух отдельных классов, GabKeyboardHook.cs и GabInterface.cs. В классе GabKeyboardHook.cs мы можем указать, какой тип ввода мы хотим обнаружить. Когда GabKeyboardHook обнаруживает ввод, он отправляет событие, содержащее информацию о обнаруженный ввод. И в классе GabInterface.cs это событие обрабатывается. Класс GabKeyboardHook.cs:

-----  
GabKeyboardHook.cs — это базовый класс GabKeyboardHook.

GabKeyboardHook.cs имеет ряд событий, которые обрабатываются на уровне приложения. Когда GabKeyboardHook получает это событие, он может обрабатывать его соответствующим образом.

GabKeyboardHook.cs сам по себе не имеет графического интерфейса или виджета, поэтому он использует класс GabInterface.cs для получения графического интерфейса и обработки событий. Класс GabInterface.cs:

----- В классе

---

GabInterface.cs мы обрабатываем все события в GabKeyboardHook.cs. Когда GabKeyboardHook получает событие, это событие транслируется в соответствующий обработчик событий в Класс GabInterface.cs. И соответствующий обработчик событий в классе GabInterface.cs зарегистрирован для обработки это конкретное событие. И все события обрабатываются этим основным классом, что делает эту библиотеку очень удобной. легко использовать. GabInterface.cs также имеет отдельный класс для графического интерфейса, который можно использовать для обработки событий графического интерфейса. Когда этот инструмент сохраняет свои настройки, он загружает последнюю версию GabKeyboardHook.cs. Этим мы можем указать только один класс для GabKeyboardHook. Если вы хотите использовать GabKeyboardHook для обнаружения ввода с клавиатуры, вам нужно создать новый экземпляр этого инструмент. Чтобы запустить этот

---

инструмент, вы можете использовать метод `dllMain` в своем приложении, а затем передать ссылку `user32.dll`.

Монтаж: ===== \*

Установите фреймворк `GabBinding`, если он еще не установлен: \*

Установите библиотеку `GabKeyboardHook`, если она еще не установлена. \* Теперь, если дважды щелкнуть `GabKeyboardHook.exe.exe`, визуальная студия должна открыться в режиме разработки. \* Когда этот инструмент открывается в режиме разработки, вы можете увидеть текстовое поле под инструментом.

---

### **System Requirements For GabKeyboardHook:**

ОС: Windows XP, Vista, 7, 8, 8.1, 10 (32 или 64 бит) Процессор: Intel Pentium 4 или новее Память: 256 МБ (приблизительно) Место на жестком диске: примерно 2,2 ГБ Видео: видеокарта с поддержкой DirectX 9 и оперативной памятью не менее 512 МБ, для лучшей производительности рекомендуется видеокарта на 256 МБ. Клавиатура: ANSI-совместимая, базовая QWERTY-клавиатура Мышь: ANSI-совместимая, устройство управления курсором (дополнительно) Разрешение экрана: