Вход в Linux по сертификату на JaCarta

Версия ПО: GNU/Linux, ID Protect для linux, OpenSSL, libpam-pkcs11, XCA

Токены: Любые

Проблема:

Настроить вход на рабочую станцию, используя сертификат с закрытым ключом, сгенерированным на стороне смарт-карты.

Решение:

Настройку входа в систему осуществляем при помощи **OpenSSL** и его графического интерфейса **XCA**. Работа на стенде с Debian testing.

Устанавливаем пакеты pcscd, libccid, openssl, xca, libengine-pkcs11-openssl, libengine-pkcs11-openssl1.1, libpam-pkcs11

Замечание: пакет **libengine-pkcs11-openssl** в репозиториях представлен в двух версиях. Устанавливались оба.

Драйвер для JaCarta PKI: <u>https://www.aladdin-rd.ru/support/downloads/989e9d58-8600-4d97-a039-e209c7c8fa8e</u>

1. Настройка ЦС.

Запускаем хса от имени root.

Замечание: вероятно, можно это делать и без привилегий суперпользователя. Но стенд настраивался именно так.

Переходим на вкладку сертификаты, создаем новый для сервера СА.

X Certificate and Key management <2>

? ~ ^ >

Источник	Владелец	Расширения	Область прим	енения ключа	Netscape	Дополнительно]
Подписа	нный запрос—	도문가가가ㅋㅋ					
Испол	њзавать подпи	ісанный <u>з</u> апрос н	а сертификат				-
🗶 Копир	овать расшире	ения из запроса			Показат	ь запрос	
Измен	нить владельца	а в запросе		Amaar			
			noc vo	преть детали			
Подписа	ние						
• Созда	ать самоподпис	санный сертифик	кат с серийным	номером 1			
	_						
O Use th	ais Cortificato fo	r cigning		wks.dbp			_
⊖ Use <u>t</u> h	nis Certificate fo	or signing		wks-dbn			Ŧ
⊖ Use <u>t</u> h	nis Certificate fo	or signing	A	wks-dbn			Ţ
O Use th	nis Certificate fo	r signing	10	wks-dbn			
○ Use <u>t</u> h лгоритм	nis Certificate fo	r signing		wks-dbn SHA 256			
○ Use <u>t</u> h	nis Certificate fo	or signing		wks-dbn			-
○ Use <u>t</u> h лгоритм •Шаблон ,	nis Certificate fo подписи для нового сер	or signing отификата ———		wks-dbn			· · · · · · · · · · · · · · · · · · ·
○ Use <u>t</u> h олгоритм •Шаблон , [default]	nis Certificate fo подписи для нового сер] СА	or signing отификата ———		wks-dbn			·
○ Use <u>t</u> h лгоритм Шаблон , [[default]	nis Certificate fo подписи для нового сер] CA	or signing отификата ———		wks-dbn			
○ Use <u>t</u> h лгоритм Шаблон , [default]	nis Certificate fo подписи для нового сер] CA	r signing отификата	Применить (wks-dbn SHA 256 расширения) Г	1рименить вл	адельца) Примен	▼
○ Use <u>t</u> h лгоритм Шаблон , [default]	nis Certificate fo подписи для нового сер] СА	or signing отификата ———	Применить (wks-dbn	Ірименить вл	адельца) Примен	 ✓ ИТЬ ВСЕ
○ Use <u>t</u> h лгоритм - Шаблон , [default]	nis Certificate fo подписи для нового сер] CA	r signing отификата ———	Применить р	wks-dbn	Ірименить вл	адельца) Примен	▼
○ Use <u>t</u> h Алгоритм - Шаблон , [default]	nis Certificate fo подписи для нового сер] СА	or signing отификата ———	Применить	wks-dbn	1рименить вл	адельца) Примен	✓ ИТЬ ВСЕ Ø ОТРАЕ

На вкладке "Источник" выбираем самоподписанный сертификат, алгоритм подписи SHA 256.

Далее идём на вкладку "Владелец":

Создание х509 сертификата



Источник Владелец	Расши	рения	Область при	менения ключа	Netsca	pe "	Дополнител	іьно
Distinguished name]
Внутреннее имя				organizationName	e [
countryName				organizationalUni	tName			
stateOrProvinceName				commonName	[wks-db	n.aladdin.ru	
localityName				emailAddress	ľ			
Тип				Содержание				Добавить
								Удалить
– Закрытый ключ ———							<i></i>	
wks-dbn.aladdin.ru (RSA	A:2048 bit	:)	🝷 🗌 Отобра	жать уже использ	зованные	е ключи	и <u>С</u> оздати	ь новый ключ
							🛛 🖋 ОК	🖉 Отмена

Заполняем необходимые поля, в поле **commonName** указываем имя хоста.

Здесь же нажимаем "Создать новый ключ".

Новый ключ		(
Введите название	и задайте тип и размер нового ключа		
Свойства ключа			
Имя ключа	wks-dbn.aladdin.ru		
Тип ключа	RSA		-
Длинна ключа	2048 bit		•
Remember as de	efault		
		🖋 Создать	🥝 Отмена
Нажимаем "Создать'	•		

Источник Владелец Расширения Область применения ключа Netscape Дополнительно X509v3 Basic Constraints	Создание х509 сертификата	a Promising the
Период действия Не раньше, чем 2017-09-20 09:23 GMT • Не позже, чем 2018-09-20 07:52 GMT • Полночь Local time Нет четко определенного срока X509v3 Subject Alternative Name X509v3 Issuer Alternative Name Pедактировать X509v3 CRL Distribution Points Authority Information Access OCSP •	Источник Владелец Расширения Область применения ключа Netscape До Х509v3 Basic Constraints Тип Центр Сертификации ▼ Длинна пути Critical	ополнительно y identifier Subject Key Identifier Authority Key Identifier
X509v3 Subject Alternative Name Редактировать X509v3 Issuer Alternative Name Редактировать X509v3 CRL Distribution Points Редактировать Authority Information Access ОСSP	Период действия Временной диапазон Не раньше, чем 2017-09-20 09:23 GMT ▼ Не позже, чем 2018-09-20 07:52 GMT ▼ Полночь □ Local time □ Нет четко	 ▼ Применить определенного срока
	X509v3 Subject Alternative Name X509v3 Issuer Alternative Name X509v3 CRL Distribution Points Authority Information Access	Редактировать Редактировать Редактировать Редактировать Редактировать

В поле "Тип" указываем "Центр сертификации". При необходимости изменяем другие параметры.

Далее нажимаем кнопку "ОК" и убеждаемся, что новый сертификат появился в списке.

Устанавливаем указатель на этот сертификат, нажимаем "Экспорт"

Certificate export	a reminute the
Name wks-dbn	
Имя файла /etc/ssl/wks-dbn.crt	
PEM Text format with headers	Формат экспорта РЕМ (*.crt)
	🖋 ОК 🛛 🧭 Отмена

Указываем каталог /etc/ssl.

Далее ПКМ на сертификате: ЦС -> Создать CRL.

Переходим на вкладку "Списки отозванных сертификатов".

Выделяем появившийся список, нажимаем "Экспорт"

Revocation list export	
Name wks-dbn	
Имя файла /etc/ssl/crl/wks-dbn.pem	
PEM Text format with headers	Формат экспорта РЕМ (*.pem) 🔻
	🛛 🔗 ОК

Каталог для экспорта /etc/ssl/crl.

Далее создаем хэши для наших данных.

Для этого понадобится скрипт **make_hash_link.sh**. Содержимое можно взять здесь: <u>https://github.com/OpenSC/pam_pkcs11/blob/master/tools/pkcs11_make_hash_link</u>

Создаем файл make hash link.sh, делаем его исполняемым:

sudo chmod +x make_hash_link.sh

Далее последовательно выполняем:

sudo ./make_hash_link.sh /etc/ssl

sudo ./make_hash_link.sh /etc/ssl/crl

2. Шаблон сертификата.

Возвращаемся в интерфейс XCA, делаем шаблон для своего сертификата. Переходим на вкладку "Шаблоны", нажимаем "Новый шаблон"

Edit XCA template



Владелец	Расширен	ния Область применения ключ		a Netscape	Дополн	нительно		
Distinguished name								
Внутренне	е имя	SC-logon		organizationName	e (ARDS		
countryNar	ne	RU		organizationalUnitName				
stateOrProv	vinceName	Moscow		commonName	(
localityNam	ne	Moscow		emailAddress	(
								(,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	Тип			Содержание				Добавить
								Удалить
- Закрытый	ключ —							
wks-dbn.a	laddin.ru (R	SA:2048 br		жать уже исполь:	зованны	е ключи	Создать	новыи ключ
							🖋 ОК	🥝 Отмена

На вкладке "Владелец" заполняем необходимые поля.

На вкладке "Расширения" указываем "Конечный пользователь"

Edit XCA template



Владелец	Расширения	Область применения ключа	Netscape [Дополнительно			
X509v3 Bas	sic Constraints —			- Key i	dentifier		
Тип	Конечный п	ользователь	-	× <u>S</u>	ubject Key Identifier		
Длинна пути 🖹 Critical 🗌 Authority Key Identifier							
Период де	йствия 2017-09-2	0 09:34 GMT 👻	диапазон	Года	- Применить		
Не позже,	чем 2027-09-2	0 09:34 GMT 👻 🗌 Полночь	Local time	Нет четко ог	пределенного срока		
X509v3 Subje	ect Alternative N	ame			Редактировать		
X509v3 Issue	er Alternative Na	me			Редактировать		
X509v3 CRL	Distribution Point	ts			Редактировать		
Authority Info	ormation Access	OCSP 💌			Редактировать		
					🖋 ОК 🛛 🥝 Отмена		

Нажимаем "ОК"

3. Настройка модуля для смарт-карт.

Для того, чтобы можно было выпускать сертификаты с ключевой парой и сертификатами прямо на токене, необходимо добавить модуль **pkcs11** в настройках. В меню Файл -> Опции, вкладка PKCS#11 provider.

Опции XCA Settings Distinguished name PKCS#11 provider	
✓ //usr//lib/libjcPKCS11-2.so.2.2.0	Добавить Remove Search
✓ OK	🔗 Отмена

Необходимо добавить библиотеку /usr/lib/libPKCS11.so.2.

Замечание: пути могут меняться в зависимости от дистрибутива и версий ПО.

4. Выпуск сертификата.

Выпускаем сертификат для пользователя.

Вкладка "Сертификаты", кнопка "Новый сертификат", вкладка "Источник"

Источник	Владелец	Расширения	Область прим	енения ключа	Netscape	Дополнительно	
-Подписан	ный запрос —						
Исполь	завать подпи	санный <u>з</u> апрос н	а сертификат				-
🕱 Копира	вать расшире	ния из запроса			Показать	ь запрос	
Измени	ить владельца	в запросе					
 Создат Use this 	ть <u>с</u> амоподпис s Certificate fo	анный сертифик r signing	ат с серийным н	юмером 1 wks-dbn			
⊖ Создат	ть <u>с</u> амоподпис s Certificate fo	анный сертифик r signing	ат с серийным н	номером 1 wks-dbn			•
 Создат Use <u>thi</u> Олгоритм по 	ть <u>с</u> амоподпис s Certificate fo одписи	анный сертифик r signing	ат с серийным н	номером 1 wks-dbn SHA 256			•
 Создат Use this Флгоритм по Шаблон д. SC-logon 	ть <u>с</u> амоподпис s Certificate fo одписи ля нового сер	анный сертифик r signing тификата	ат с серийным н	юмером 1 wks-dbn SHA 256			▼
 Создат Use <u>t</u>hi: Олгоритм п Шаблон д SC-logon 	ть <u>с</u> амоподпис s Certificate for одписи ля нового сер	анный сертифик r signing тификата	ат с серийным н	номером 1 wks-dbn SHA 256 асширения П	ірименить вл	адельца) Примен	▼

Выбираем "Use this Certificate for sighning", алгоритм подписи SHA 256, выбираем созданный ранее шаблон. Нажимаем "Применить всё".

Идём на вкладку "Владелец":

A Transient

Создание х509 сертификата



Источник	Владелец	Расши	рения	Область при	менения ключа	Netsca	pe ,	Дополнительно	
_ Distinguish	ed name —]
Внутренне	е имя				organizationName	e 🚺	ARDS		
countryNa	me (RU			organizationalUni	tName			
stateOrPro	vinceName	Moscow			commonName		test		
localityNan	ne (Moscow			emailAddress	ľ			
					,				
	Тип				Содержание				Добавить
									Удалить
– Закрытый	ключ —]
wks-dbn a	laddin ru (BS	A-2048 bit	+)		жать уже использ	зованные	• ключ	И Создать но	вый ключ
	wks-dbn.aladdin.ru (RSA:2048 bit) 🔽 🗌 Отображать уже использованные ключи 🔼 Создать новый ключ								
								Second Contemporation of the second s	🧭 Отмена

Заполняем необходимые поля. В поле **commonName** указываем имя нужного пользователя.

Далее нажимаем "Создать новый ключ".

Новый ключ	
Введите название	и задайте тип и размер нового ключа
Свойства ключа-]
Имя ключа	test
Тип ключа	My token #0C50000427129613 (RSA Key of 1024 - 2048 bits) 🔻
Длинна ключа	2048 bit 💌
Remember as de	fault 🖋 Создать 🙋 Отмена

В поле "Тип ключа" выбираем наш токен, алгоритм RSA. Нажимаем "Создать". После ввода пин-кода будет сгенерирована ключевая пара.

Далее нажимаем "ОК", соглашаемся, чтобы сертификат был сохранён на токене.

Теперь этот токен можно использовать для входа в систему.

5. Настройка pam.d и pam_pkcs11.

Для возможности входа по токену в pam.d необходимо добавить модуль pam pkcs11.

Настраивается модуль в файле /etc/pam_pkcs11/pam_pkcs11.conf. Если файла нет, пример его можно взять по ссылке: <u>https://github.com/OpenSC/pam_pkcs11/blob/master/etc/pam_pkcs11.conf.example.in</u>

В секцию **pam_pkcs11** добавляем модуль JaCarta:

```
pkcsll_module JaCarta {
    module = /usr/lib/libjcPKCS11-2.so;
    description = "JaCarta PKCS#11 module";
    slot_num = 0;
    support_threads = true;
    ca_dir = /etc/ssl;
    crl_dir = /etc/ssl/crl;
    cert_policy = ca,signature; }
```

Замечание: пути могут меняться в зависимости от дистрибутива и версий ПО.

Меняем параметр:

use_pkcs11_module = JaCarta;

Меняем файл pam.d. Находим в каталоге /etc/pam.d файл вашего менеджера, например lightdm. В начало файла вставим строку:

auth sufficient pam_pkcs11.so config_file=/etc/pam_pkcs11/pam_pkcs11.conf

Замечание: если мы хотим исключить другие методы аутентификации, меняем директиву **sufficient** на **required**. Не рекомендуется экспериментировать на файлах muna login, common-auth и m.д.

Далее при подключенном токене при входе в систему ввести пин-код (некоторые DM выдают явное сообщение, что нужно авторизоваться именно на токене) и войти в систему.

ID статьи: 227

Последнее обновление: 18 Oct, 2017

Ревизия: 1

JaCarta -> Вход в Linux по сертификату на JaCarta

https://kbp-6.aladdin-rd.ru/index.php?View=entry&EntryID=227