



Universidad Nacional de Asunción  
**CNC**  
Centro Nacional de Computación

**NIC.py**

**32** AÑOS  
1991-2023  
**.PY** te conecta  
al mundo

Universidad  
Católica  
"Nuestra Señora de la Asunción"

**LE**  
LABORATORIO  
de Electrónica  
Digital

# PARAGUAY 2023

## INTERNET GOVERNANCE FORUM

 FORO DE GOBERNANZA  
DE INTERNET PARAGUAY

[www.nic.py](http://www.nic.py)

2 y 3 de octubre/2023

1



# DNSSEC

## para .py

Conceptos y práctica experimental

# AGENDA

- Repaso de conceptos generales: DNS y DNSSEC
- Propuesta para implementación
- Procedimiento de encadenamiento

# REPASO

## DNSSEC y criptografía

### Tres conceptos clave

- Claves públicas / privadas
- Mensajes “digests”, sumas de comprobación, hashes.
- Firmas digitales

Están en el núcleo de DNSSEC. Si estos no tienen sentido, entonces DNSSEC no tendrá sentido.

# REPASO

## Texto cifrado

- Comenzamos con **texto plano**. Algo que puedas leer.
- Aplicamos un algoritmo matemático al **texto plano**.
- El algoritmo es el **cifrado**.
- El texto plano se convierte en **texto cifrado**.
- Crear un cifrado seguro es un proceso difícil.
- El proceso de estandarización para AES, el reemplazo para el envejecimiento del protocolo DES, tomó 5 años.

# REPASO

## Llaves

- En la *criptografía simétrica*, un texto simple se transforma en un *texto cifrado*, y nuevamente en *texto plano* utilizando una clave para el cifrado (el algoritmo utilizado) en ambos extremos.
- Suponiendo que se conoce el método de cifrado, la seguridad del *texto cifrado* se basa en la *llave*. Éste es un punto crítico. Si alguien obtiene su *llave*, se compromete su *texto plano*.

# REPASO

## Cifrado simétrico

### Llave única/Cifrados simétricos



La misma clave se utiliza para cifrar el documento antes de enviarlo y para descifrarlo una vez que se recibe



# REPASO

Cifrado simétrico

**Llave única/Cifrados simétricos**

Problema: ¿Cómo hacer para distribuir de manera segura la clave a todas las partes destinadas?



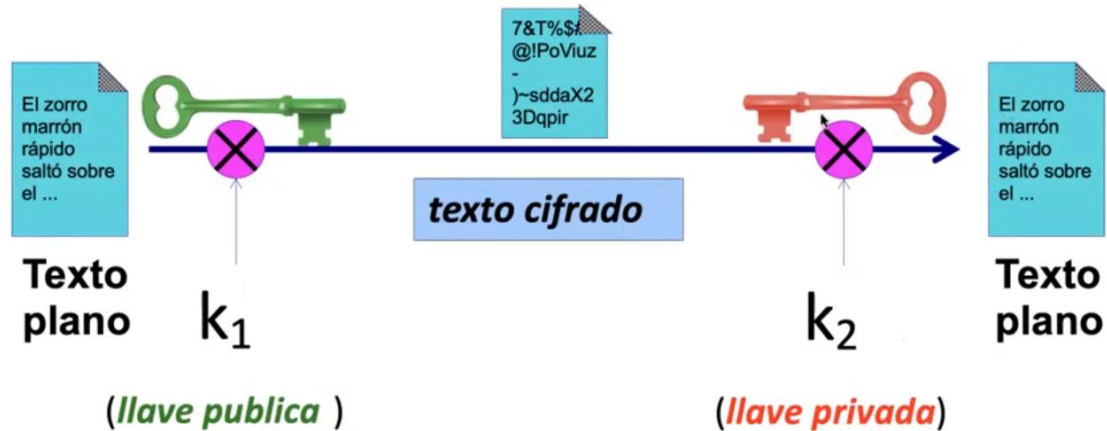
# REPASO

## Llaves públicas/privadas

- Generamos un par de llaves de cifrado. Una llave es la *llave privada*, la otra es la *llave pública*.
- La *llave privada* permanece secreta y debe ser protegida.
- La *llave pública* es de libre distribución. Está relacionada matemáticamente con la *llave privada*, pero no se puede (fácilmente) derivar la *llave privada* de la *llave pública*.
- Utilice la *llave pública* para cifrar los datos. Solo alguien con la *llave privada* puede descifrar los datos cifrados.

# REPASO

Ejemplo de par de llaves públicas/ privadas



Una llave se utiliza para cifrar el documento,  
una llave diferente se utiliza para descifrarlo.

**¡Este es un aspecto importante!**

[socium](https://www.socium.com)

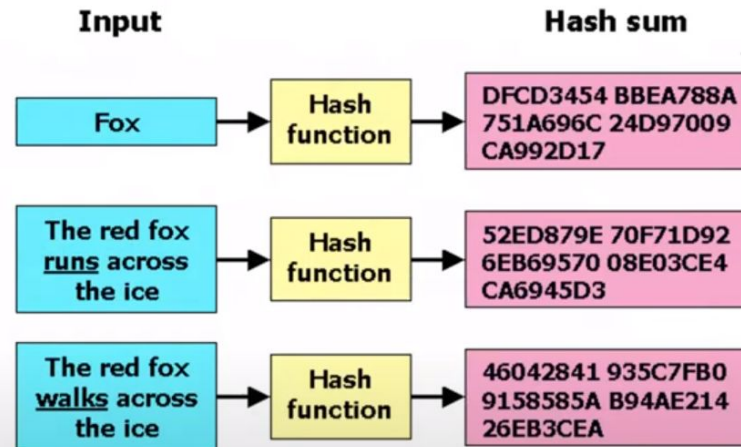
# REPASO

## Funciones de hash unidireccionales

- Una función matemática que genera un resultado de longitud fija independientemente de la cantidad de datos que pase a través de él. Generalmente muy rápido.
- No puede generar los datos originales a partir del resultado de longitud fija, por lo tanto, el término "unidireccional".
- Es de esperar que no pueda encontrar dos conjuntos de datos que produzcan el mismo resultado de longitud fija. Si lo haces, esto se llama una colisión. (Ejemplo, md5).
- El resultado de longitud fija se conoce como **un resumen de mensaje o una suma de comprobación o un hash.**

# REPASO

## Hash otro ejemplo



Tenga en cuenta el cambio significativo en la suma de hash para cambios menores en la entrada. Tenga en cuenta que la suma de hash es la misma longitud para diferentes tamaños de entrada. Esto es extremadamente útil.

\*Image courtesy Wikipedia.org.



# REPASO

¿Para qué sirve esto?

**Hay varios:**

- Cifrado de contraseñas (en Linux, Unix y Windows), utilizando múltiples rondas de hashing (MD5 u otro).
- Puede ejecutar muchos megabytes de datos a través de una función de hash, pero solo tiene que verificar un número fijo de bits de información (160-512 bits). Esto se utiliza para crear una *firma digital*.

# REPASO

## Firmas digitales

Revertir el papel de las claves públicas y privadas

Para crear una firma digital en un documento haga:

→ *Hash* del documento, produciendo un *resumen del mensaje*

1. Cifre el *resumen del mensaje* con su *llave privada*.

→ Envíe el documento más el *resumen del mensaje* cifrado.

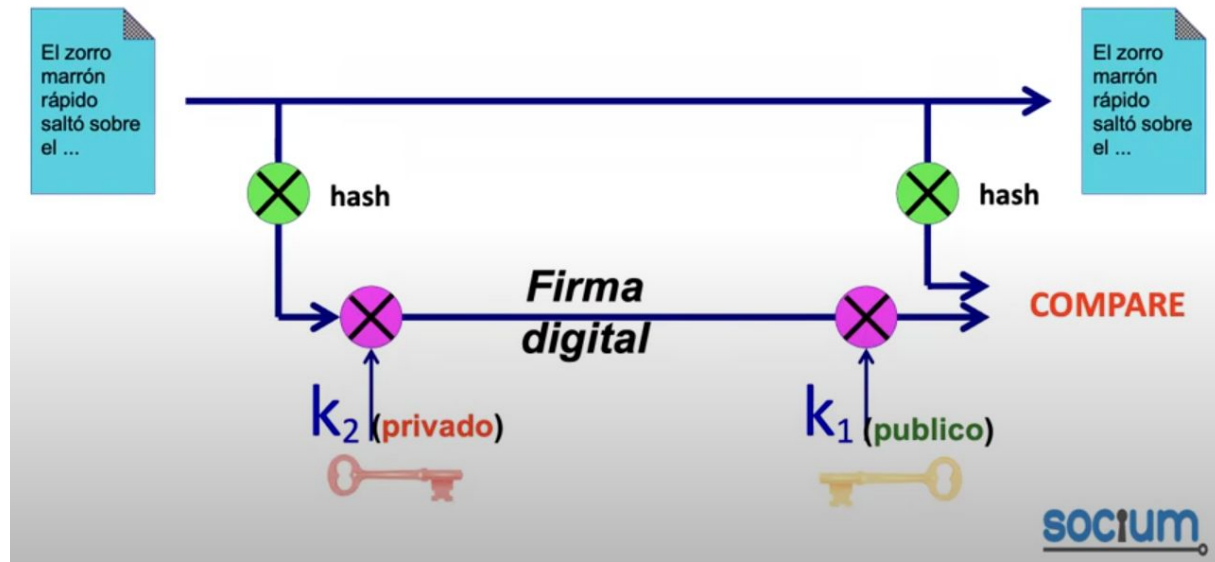
→ En el otro extremo *hash* del documento y descifre el Mensaje cifrado con la *llave pública* de la persona.

- ✓ Si los resultados coinciden, el documento es autenticado.
- ✓ Este proceso crea una *firma digital*.

# REPASO

Al autenticar:

Toma un hash del documento y encripta solo eso. Un hash encriptado se llama "firma digital"



# REPASO

## Conclusión

- Llaves públicas/privadas.
- Síntesis de mensajes, sumas de comprobación, hashes.
- Firmas digitales.

Están en el núcleo de DNSSEC.



# REPASO

- El formato de zona de BIND es muy común, así que lo usaremos aquí:

```
zone. SOA nsX.zone. hostmaster.zone.  
        ( 2009022401 ; serial  
          1d         ; refresh  
          12h        ; retry  
          1w         ; expire  
          1h )       ; neg. TTL  
  
zone.      NS  ns.zone.  
        NS  ns.otherzone.  
  
zone.      MX  5 server.otherzone.  
www.zone.  A   1.2.3.4
```



UNIVERSITY OF OREGON



# REPASO

- Estructura de los records:

NAME	[TTL]	TYPE	DATA (type specific)
-----			
host.zone.	3600	A	10.20.30.40
sub.zone.	86400	MX	5 server.otherzone.





UNIVERSITY OF OREGON



# REPASO

- Múltiples records con *el mismo nombre y tipo* se agrupan en Resource Record Sets (RRsets):

<pre>mail.zone.      MX mail.zone.      MX</pre>	<pre>5 server1.zone. 10 server2.zone.</pre>	} RRset
<pre>server1.zone.  A server1.zone.  A server1.zone.  A</pre>	<pre>10.20.30.40 10.20.30.41 10.20.30.42</pre>	} RRset
<pre>server1.zone.  AAAA server1.zone.  AAAA</pre>	<pre>2001:123:456::1 2001:123:456::2</pre>	} RRset
<pre>server2.zone.  A</pre>	<pre>11.22.33.44</pre>	} RRset



# REPASO

## DNSSEC – breve resumen

- Autenticidad de datos e integridad al firmar los RRSets con una clave **privada**
- Claves **públicas** (DNSKEYS), para verificar las firmas (RRSIGs)
- Los sub-dominios firman sus zonas con su llave **privada**
  - La autenticidad de la llave se establece gracias a la firma/checksum del record delegation signer (DS) por la zona superior
- Repetir en la zona superior...
- No es tan difícil en papel
  - Operativamente, es un poco más complicado



# REPASO

## DNSSEC – breve resumen

- Autenticidad de datos e integridad al firmar los RRSets con una clave **privada**
- Claves **públicas** (DNSKEYS), para verificar las firmas (RRSIGs)
- Los sub-dominios firman sus zonas con su llave **privada**
  - La autenticidad de la llave se establece gracias a la firma/checksum del record delegation signer (DS) por la zona superior
- Repetir en la zona superior...
- No es tan difícil en papel
  - Operativamente, es un poco más complicado



# REPASO

## DNSSEC - Conceptos

- Cambia el modelo de confianza de DNS de “abierto” y “de confianza” a uno de “verificable”
- Uso extensivo de criptografía asimétrica para lograr:
  - Autenticación del origen
  - Integridad de los datos
  - Autenticidad de la negación de existencia
- No se trata de proveer confidencialidad
- DNSSEC no implica una carga computacional en los servidores autorizados ( != los *firmantes* )
- No se cambia esencialmente el protocolo
  - Puede coexistir con la infraestructura actual



- ... Bueno, más o menos (EDNS0)

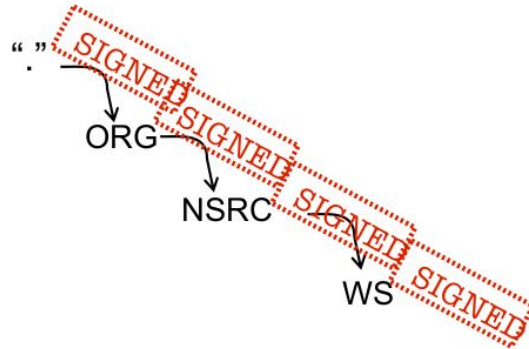
UNIVERSITY OF OREGON



# REPASO

## DNSSEC - conceptos

- Crear una **cadena de confianza** usando el modelo existente basado en delegaciones para la distribución que es el DNS
- No se firma la zona completa, se firma un RRset



- Nota: la zona superior NO firma la zona inferior.
  - La superior firma un *apuntador* (hash) a la *clave* usada para firmar los datos en la zona inferior (importante!)



UNIVERSITY OF OREGON



# REPASO

## DNSSEC: nuevos RRs

Se añaden cuatro nuevos Resource Records\*:

1. **DNSKEY**: Llave pública utilizada en el proceso de firmado.
2. **RRSIG**: Firma de un RRset
3. **NSEC/NSEC3**: Provisto como evidencia de que el nombre y/o el tipo de RR no existe
4. **DS**: Delegation Signer. Contiene el *hash* de la clave pública usada para firmar la llave que a su vez servirá para firmar los datos de la zona. Se siguen los RRs DS hasta encontrar una zona “de confianza” (idealmente la raíz).

\*Vea la excelente discusión de Geoff Huston en <http://ispcolumn.isoc.org/2006-08/dnssec.html>



UNIVERSITY OF OREGON





# REPASO

## DNSSEC: KSK y ZSK

- Para permitir la renovación de claves (“rollovers”), se generan dos pares de claves:
  - **Key Signing Key (KSK)**
    - Referenciada por la zona superior (Secure Entry Point), en forma de DS (Delegation Signer)
    - Usada para firmar la Zone Signing Key (ZSK)
  - **Zone Signing Key (ZSK)**
    - Firmada por la Key Signing Key
    - Usada para firmar los RRsets
- Esta disociación permite la renovación del ZSK sin tener que renovar la KSK (y el DS en la superior) – menos interacción administrativa

# REPASO

## DNSSEC: DS

- Dos hashes generados por defecto:
  - 1 SHA-1 MANDATORY
  - 2 SHA-256 MANDATORY
- Hay nuevos algoritmos que están en proceso de estandarización
- Esto ocurrirá continuamente a medida que los algoritmos se determinen inseguros

# REPASO

---

## Firmar la zona (utilizando las herramientas BIND)

1. Generar los pares de claves
2. Incluir los récords DNSKEYs públicos en el contenido de la zona
3. Firmar la zona con la llave secreta ZSK
4. Publicar la zona
5. Enviar los récords DS a la zona superior
6. Esperar...



# REPASO

## 1. Generar las claves

```
# Generar la ZSK  
dnssec-keygen [-a rsasha1 -b 1024] -n ZONE myzone
```

```
# Generar la KSK  
dnssec-keygen -a [rsasha1 -b 2048] -n ZONE -f KSK  
myzone
```

Esto generó 4 ficheros:

```
Kmyzone.+005+id_of_zsk.key  
Kmyzone.+005+id_of_zsk.private  
Kmyzone.+005+id_of_ksk.key  
Kmyzone.+005+id_of_ksk.private
```



```
root@vm-taller:/var/lib/bind# dnssec-keygen -a ECDSAP256SHA256 -b 1024 udnsec30-taller.com.py.hosts  
Generating key pair.  
Kudnssec30-taller.com.py.hosts.+013+13536  
root@vm-taller:/var/lib/bind# dnssec-keygen -a ECDSAP256SHA256 -b 1024 -f KSK udnsec30-taller.com.py.hosts  
Generating key pair.  
Kudnssec30-taller.com.py.hosts.+013+41145
```

# REPASO

## 2. Incluir las claves en la zona

Incluya los records DNSKEY para la ZSK y KSK en la zona, para que sean firmados con el resto de los datos:

```
cat Kmyzone*key >>myzone
```

O utilice la instrucción \$INCLUDE para que sean cargados al leer la zona:

```
$INCLUDE "Kmyzone.+005+id_of_zsk.key"
```

```
$INCLUDE "Kmyzone.+005+id_of_ksk.key"
```

# REPASO

## 3. Firmar la zona

Firme su zona

```
# dnssec-signzone myzone
```

- *dnssec-signzone* usará todos los valores por defecto para la duración de la firma, el número de serie no será incrementado, y las claves privadas para firmar serán determinadas automáticamente.
- Firmar hará lo siguiente:
  - Ordenar los datos (lexicográficamente)
  - Insertar:
    - Récor ds NSEC
    - Récor ds RRSIG (firma de cada RRset)
    - Récor ds DS de ficheros *key-set* de sub-zonas (para la zona superior)
      - Generar ficheros *key-set* y *DS-set*, para enviar a la zona



# REPASO

## 3. Firmar la zona (2)

- ISC BIND
- Desde la versión 9.7.0, BIND puede firmar y re-firmar sus zonas automáticamente
  - Facilita las cosas significativamente
  - Pero la generación de las claves, la gestión y la renovación aún tienen que hacerse por separado.
- La versión 9.8.0 introduce la firma en línea
  - Integración más sencilla en la cadena de producción existente.

# REPASO

## 4. Publicar las zonas firmadas

- Para publicar la zona firmada es necesario configurar el servidor para que cargue el fichero de la zona firmada.
- ... pero usted aún tiene que enviar sus récords DS de manera segura a la zona superior, de lo contrario, nadie sabrá que usted está firmando su zona con DNSSEC



# REPASO

## 4. Publicar las zonas firmadas

```
mfredes@mfredes:~$ dig @8.8.8.8 udnsssec-taller.com.py ANY
; <<>> DiG 9.16.1-Ubuntu <<>> @8.8.8.8 udnsssec-taller.com.py ANY
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 49906
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 16, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;udnsssec-taller.com.py.      IN      ANY

;; ANSWER SECTION:
udnsssec-taller.com.py. 3600   IN      DNSKEY 256 3 13 JfIkIf/E2hV8J30uC2+U8M4tB6UZQ/LWT8Pkf9nIv/LNSEPU2wt6fazr XaTex00h14x+HrQPQ1N1PymNFDCjHg==
udnsssec-taller.com.py. 3600   IN      DNSKEY 257 3 13 xe7ji9bJgC06lnsQn/qgmQLMw2jMJHnDNmI8TthVB8sap+Shq7vTLXxc 0+/qxoCJ8cNdHxb5HbIadrjhk8DBA==
udnsssec-taller.com.py. 3600   IN      RRSIG  SOA 13 3 3600 20231101134237 20231002134237 12012 udnsssec-taller.com.py. 3JNh2wylsbaBGiGXXlabmEgDGATf74v91gFxDwDcX47neg9nyT29egQ /dTyy9STb0WJBLABWP0gV7QtT5d0Zg==
udnsssec-taller.com.py. 3600   IN      RRSIG  NS 13 3 3600 20231101132947 20231002132947 12012 udnsssec-taller.com.py. ZosoJ0iDfe6TNMES12T/d+I0+reA8CBbWtl1axQp0cBFJQe0U5qGbKu v0yNzmxR67XCqpsRXKwdsV2S0mX4dA==
udnsssec-taller.com.py. 3600   IN      RRSIG  A 13 3 3600 20231101035450 20231002035450 12012 udnsssec-taller.com.py. ln/0yam9shAsy0M9K9BI0TVzZxFxU/ZbaLI6Wksjt7dGHGz8d51vbVSM vzf5QV7J+sTVHDt58EGA1iZTCVgFQ==
udnsssec-taller.com.py. 3600   IN      RRSIG  AAAA 13 3 3600 20231101035450 20231002035450 12012 udnsssec-taller.com.py. Re7429YUw0baRp1EM5Q/2jB2ab2RhwW58JG5Zf84UIjuFbjzU94M0Yb3 f3Z6e+PqAretjxt6wfvL7+M/DmrUA==
udnsssec-taller.com.py. 3600   IN      RRSIG  DNSKEY 13 3 3600 20231101035450 20231002035450 12012 udnsssec-taller.com.py. 9x+f8/Nyixds7haIlpHdcRbPz8HQ17C8GbhG03Sj15DDaV4qoiwvkEog aMUMxZmyXchAdup6Hjcr6SKd+U+ffg==
udnsssec-taller.com.py. 3600   IN      RRSIG  DNSKEY 13 3 3600 20231101035450 20231002035450 53845 udnsssec-taller.com.py. OpuGcLFkyK9yNow4QIZGd+BMdNlXDD8Rv/WLFG9xulB9U1DBRHEnpHsq /dZi5FamL4mqEGCxlYnfc d+Ab3Lfg==
udnsssec-taller.com.py. 0       IN      NSEC3PARAM 1 0 0 -
udnsssec-taller.com.py. 0       IN      RRSIG  NSEC3PARAM 13 3 0 20231101035450 20231002035450 12012 udnsssec-taller.com.py. bf8/CEdPMQLpajI08d2VW3RLFxzqbxUEHP1pZQ9QHRNm5NxmD0DQn6 5QLkS/IGMBx8wACw9zaL2GvgZGSjdQ==
udnsssec-taller.com.py. 3600   IN      SOA    dns-firmador-taller.nic.py. email.udnsssec-taller.com.py. 2023100215 3600 600 1209600 3600
udnsssec-taller.com.py. 3600   IN      NS     dns-firmador-taller.nic.py.
udnsssec-taller.com.py. 3600   IN      A      200.10.228.61
udnsssec-taller.com.py. 3600   IN      A      200.10.228.60
udnsssec-taller.com.py. 3600   IN      A      200.10.228.62
udnsssec-taller.com.py. 3600   IN      AAAA   2001:1320:f000::228:131

;; Query time: 152 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: lun oct 02 13:20:53 -03 2023
;; MSG SIZE rcvd: 1202
```

## 4. Publicar las zonas firmadas

<https://digwebinterface.com/?hostnames=udnssec-taller.com.py&type=ANY&userresolver=8.8.4.4&ns=auth&nameservers=>

The screenshot shows the digwebinterface.com website with the following configuration:

- Hostnames or IP addresses: `udnssec-taller.com.py`
- Type: `ANY`
- Nameservers: `Resolver: Default`
- Options: `Authoritative` (selected), `NIC`, `Specify myself`

The output shows a truncated DNS query result for `udnssec-taller.com.py@dns-firmador-taller.nic.py:`

```
;; Truncated, retrying in TCP mode.
udnssec-taller.com.py. 3600 IN DNSKEY 256 3 13 JfIkIf/E2hV8J30uC2+U8M4tB6UZ0/LWT8Pkf9niV/LNSEPU2wt6fazR XaTex00h14x+HrQP0I1N1PymnFDCjHg==
udnssec-taller.com.py. 3600 IN DNSKEY 257 3 13 xe7ji9bJgC06lns0n/qgmQLMw2jMJHnDNM18TthVB8sap+5hq7VtLxc 0+/qxc0J8cNdhbxb5HbIadrjhk8DBA==
udnssec-taller.com.py. 3600 IN RRSIG SOA 13 3 3600 20231101134237 20231002134237 12012 udnssec-taller.com.py. 3JNh2wyLsbaBGiGXXLbmEgDGAtF74v9lgFxDwdcX47neg9nyT29eq0 /dTYy9STb0WJBLABWP0gV70tT5d0zg==
udnssec-taller.com.py. 3600 IN RRSIG NS 13 3 3600 20231101132947 20231002132947 12012 udnssec-taller.com.py. ZosoJ0iDf6e6TNMEs12T/d+I0+reA8CBbWtllaxQYp0cBFJQe0U5qGbkU v0yNzmxR67xCqpsRXKwdsv250mx4dA==
udnssec-taller.com.py. 3600 IN RRSIG A 13 3 3600 20231101035450 20231002035450 12012 udnssec-taller.com.py. ln/0yam9shA5Yom9K9BI0TVzZQfXU/ZbaLI6Wksjt7dGHGz8d51vbVSM vzf5QV7J+sTVHdT58eEGAi1ZTCVgT0==
udnssec-taller.com.py. 3600 IN RRSIG AAAA 13 3 3600 20231101035450 20231002035450 12012 udnssec-taller.com.py. Re7429YUwObaRp1EM5Q/2jB2ab2RhwW58JG5Zf84UjjuFbjzU94M0Yb3 f326e+PgAretjxt6wfvL7+M/DmrmUA==
udnssec-taller.com.py. 3600 IN RRSIG DNSKEY 13 3 3600 20231101035450 20231002035450 12012 udnssec-taller.com.py. 9x+f8/Ny1xd57haIlpWdcRbPz8HQ17C8GbwG035j15DDaV4goiwwkEdg aMUMxZmyXchAdup6Hjcr6SKd+U+ffGg==
udnssec-taller.com.py. 0 IN RRSIG DNSKEY 13 3 3600 20231101035450 20231002035450 53845 udnssec-taller.com.py. 0puqCLfkyK9yNow4QIZGd+BMdNlXDD8Rv/WLFG9xulB9U1D8RHEnpHsq /dzI5FamL4mqEGCxlYnfcjcd+Ab3Lfg==
udnssec-taller.com.py. 0 IN NSEC3PARAM 1 0 0 -
udnssec-taller.com.py. 3600 IN RRSIG NSEC3PARAM 13 3 0 20231101035450 20231002035450 12012 udnssec-taller.com.py. bf8/CedPMQLpajI08d2WV3RFLXfzqbxUEHP1pZ0g9QHRNm5NxzmD00Qn6 5QLkS/IGMBx8wACw9zal2GvgZG5jd0==
udnssec-taller.com.py. 3600 IN SOA dns-firmador-taller.nic.py. email.udnssec-taller.com.py. 2023100215 3600 600 1209600 3600
udnssec-taller.com.py. 3600 IN NS dns-firmador-taller.nic.py.
udnssec-taller.com.py. 3600 IN A 200.10.228.62
udnssec-taller.com.py. 3600 IN A 200.10.228.60
udnssec-taller.com.py. 3600 IN A 200.10.228.61
udnssec-taller.com.py. 3600 IN AAAA 2001:1320:f000::228:131
```

# REPASO

## Caducidad de las firmas

- Las firmas caducan en 30 días por defecto (BIND)
- Es necesario firmar regularmente:
  - Para mantener una ventana constante de validez para las firmas de un RRset *existente*
- Para firmar *RRsets nuevos y actualizados*
- Quién hace esto ?
- Las claves en sí NO caducan...
  - Pero sí necesitan ser renovadas...



# REPASO

## Caducidad de las firmas

- Las firmas caducan en 30 días por defecto (BIND)
- Es necesario firmar regularmente:
  - Para mantener una ventana constante de validez para las firmas de un RRset *existente*
- Para firmar *RRsets nuevos y actualizados*
- Quién hace esto ?
- Las claves en sí NO caducan...
  - Pero sí necesitan ser renovadas...



# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN

## **BIND**

```
#sudo apt-get upgrade  
#sudo apt install -y bind9 bind9utils bind9-doc dnstools
```

## **WEBMIN**

```
# curl -o setup-repos.sh https://raw.githubusercontent.com/webmin/webmin/master/setup-repos.sh  
# sh setup-repos.sh  
# apt-get install webmin --install-recommends
```

## **Recursos para descarga**

<https://www.nic.py/dnssec>

## **Ref.**

<https://webmin.com/>

# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN

DNSSEC Key Re-Signing

Zones signed with DNSSEC typically have two keys - a zone key which must be re-generated and signed regularly, and a key signing key which remains constant. This page allows you to configure Webmin to perform this re-signing automatically.

Key re-signing options

Automatic key re-signing enabled?  
 Yes  No

Period between re-signs?  
21 days

Save

Return to zone list

Periodo de re-firmado < 30 días

# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN

## PRÁCTICA

- Generar las llaves
- Firmar y publicar la zona
- Realizar análisis preliminar
- Realizar el encadenamiento con la Zona Superior “.PY”

# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN

The screenshot displays the 'Setup DNSSEC Key' interface for the zone 'udnssec-taller.com.py'. The page title is 'Setup DNSSEC Key' and the URL is 'udnssec-taller.com.py'. A message states: 'This zone does not have a DNSSEC signing key yet. You can use this form to have Webmin create one, so that clients resolving this zone are protected against DNS spoofing attacks.'

The 'New DNSSEC key options' section includes:

- Key algorithm:** A dropdown menu with options: ECDSAP256SHA256, RSASHA1, **RSASHA256** (highlighted in red), RSAMD5, DSA, DH, HMAC-MD5, NSEC3RSASHA1, NSEC3DSA, ECDSAP256SHA256, and ECDSAP384SHA384.
- Key size:** Radio buttons for 'Average size' (selected), 'Strongest allowed', and 'Other size (in bits)' with an input field.
- Number of keys to create:** Radio buttons for 'Zone key and key-signing key' (selected) and 'Zone key only'.

Below the form, a progress bar shows the following steps:

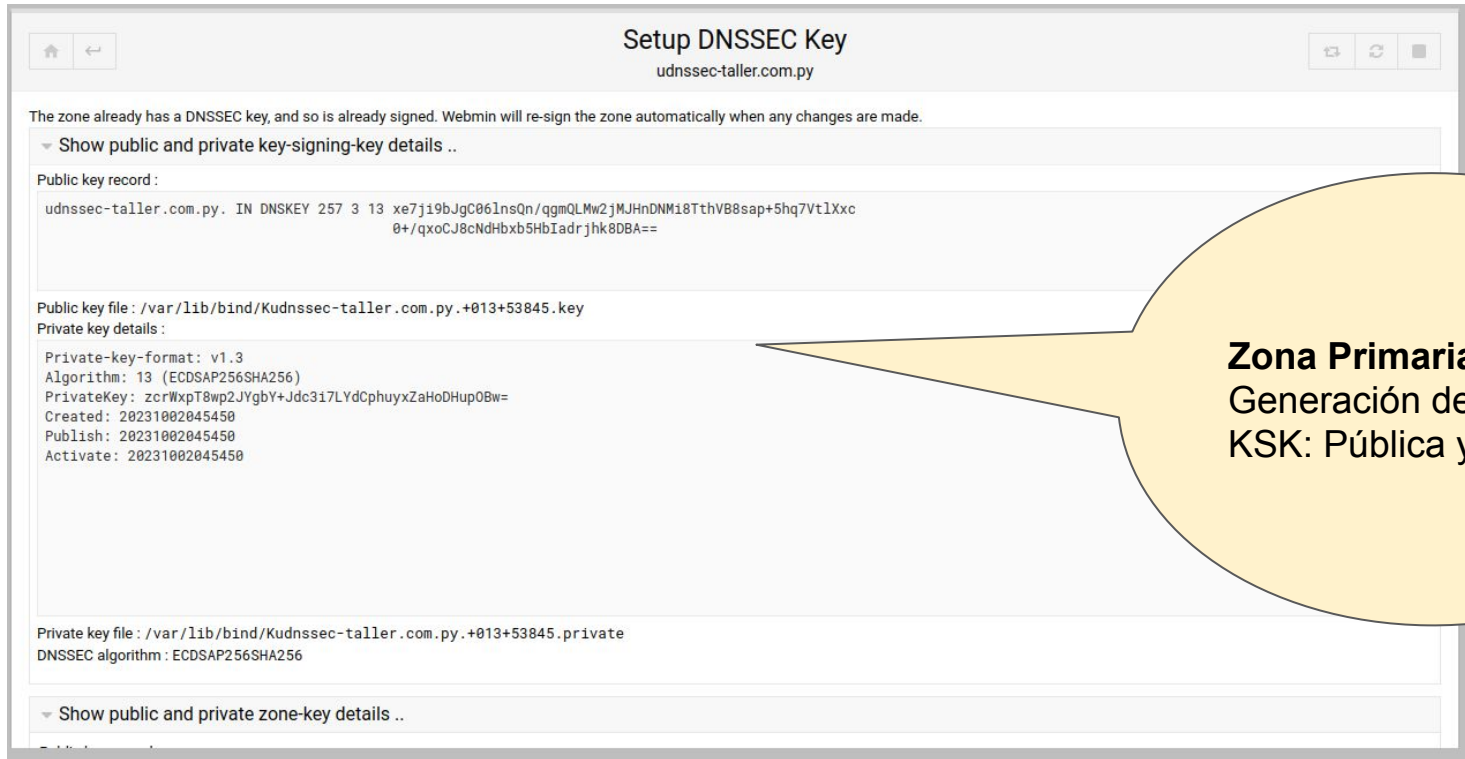
- Creating DNSSEC key for udnssec-taller.com.py ..
- .. done
- Signing zone udnssec-taller.com.py with new key ..
- .. done

A yellow callout bubble on the right contains the text: **Zona Primaria**  
Generación de llaves KSK y ZSK.



# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN



**Setup DNSSEC Key**  
udnssec-taller.com.py

The zone already has a DNSSEC key, and so is already signed. Webmin will re-sign the zone automatically when any changes are made.

▼ Show public and private key-signing-key details ..

Public key record :

```
udnssec-taller.com.py. IN DNSKEY 257 3 13 xe7j19bJgC061nsQn/qgmQLMw2jMJHnDNM18TthVB8sap+5hq7Vt1Xxc
0+/qxoCJ8cNdHbxb5HbIadrjkh8DBA==
```

Public key file : /var/lib/bind/Kudnssec-taller.com.py.+013+53845.key

Private key details :

```
Private-key-format: v1.3
Algorithm: 13 (ECDSAP256SHA256)
PrivateKey: zcrWxpT8wp2JYgbY+Jdc3i7LYdCphuyxZaHoDHup0Bw=
Created: 20231002045450
Publish: 20231002045450
Activate: 20231002045450
```

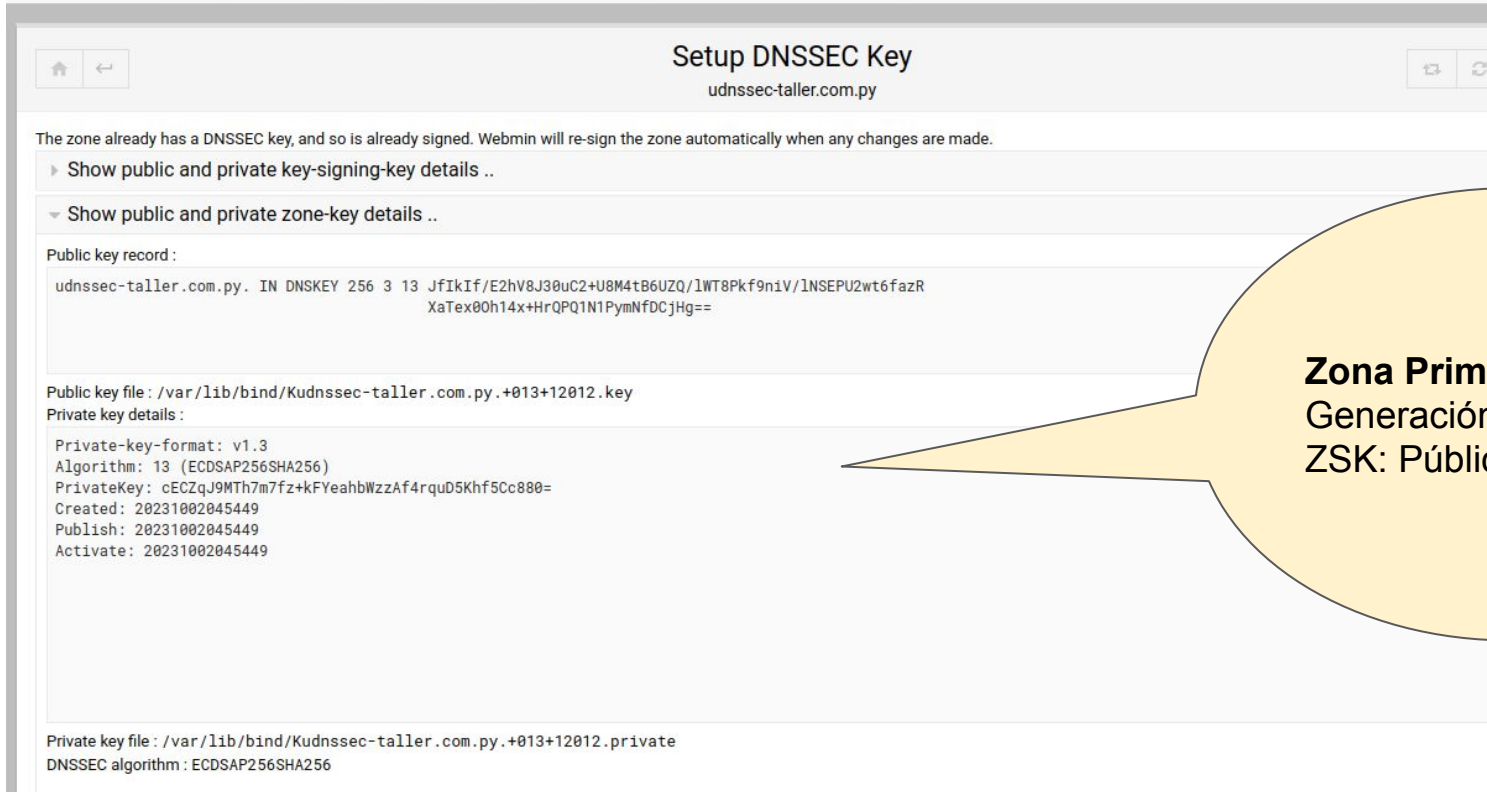
Private key file : /var/lib/bind/Kudnssec-taller.com.py.+013+53845.private  
DNSSEC algorithm : ECDSAP256SHA256

▼ Show public and private zone-key details ..

**Zona Primaria**  
Generación de llaves  
KSK: Pública y Privada

# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN



Setup DNSSEC Key  
udnsssec-taller.com.py

The zone already has a DNSSEC key, and so is already signed. Webmin will re-sign the zone automatically when any changes are made.

▶ Show public and private key-signing-key details ..

▼ Show public and private zone-key details ..

Public key record :

```
udnsssec-taller.com.py. IN DNSKEY 256 3 13 JfIkIf/E2hV8J30uC2+U8M4tB6UZQ/IWT8Pkf9n1V/INSEPU2wt6fazR
XaTex00h14x+HrQPQ1N1PymNfDCjHg==
```

Public key file : /var/lib/bind/Kudnssec-taller.com.py.+013+12012.key

Private key details :

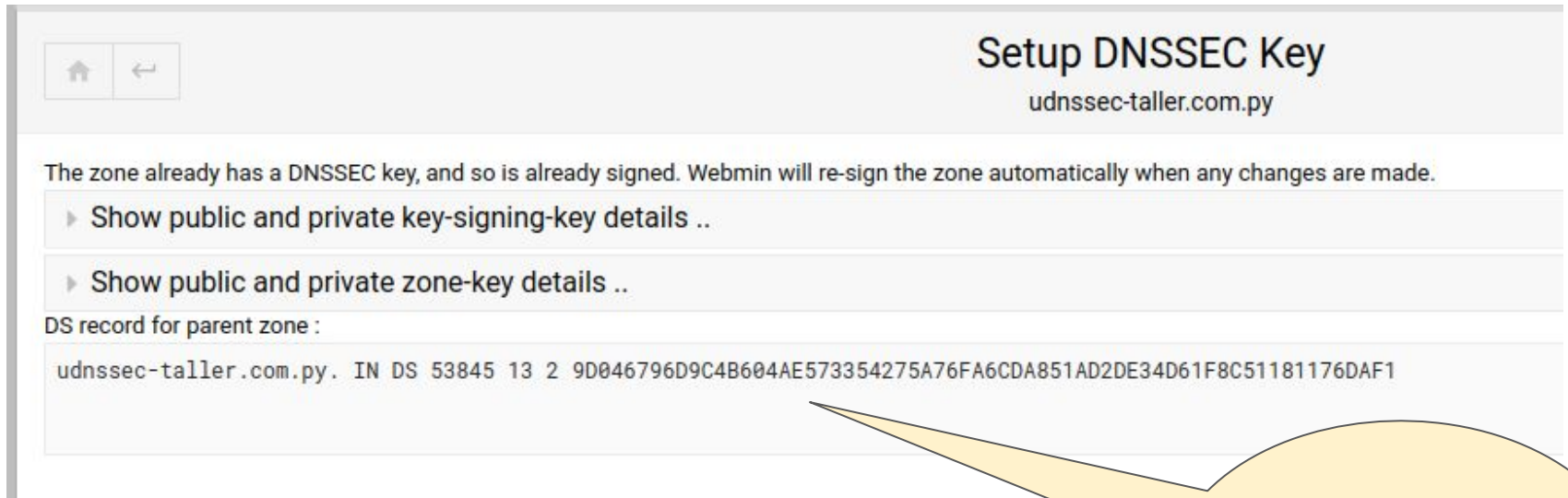
```
Private-key-format: v1.3
Algorithm: 13 (ECDSAP256SHA256)
PrivateKey: cECZqJ9MTh7m7fz+kFYeahbWzzaF4rquD5Khf5Cc880=
Created: 20231002045449
Publish: 20231002045449
Activate: 20231002045449
```

Private key file : /var/lib/bind/Kudnssec-taller.com.py.+013+12012.private  
DNSSEC algorithm : ECDSAP256SHA256

**Zona Primaria**  
Generación de llaves  
ZSK: Pública y Privada

# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN



Setup DNSSEC Key  
udnssec-taller.com.py

The zone already has a DNSSEC key, and so is already signed. Webmin will re-sign the zone automatically when any changes are made.

- ▶ Show public and private key-signing-key details ..
- ▶ Show public and private zone-key details ..

DS record for parent zone :

```
udnssec-taller.com.py. IN DS 53845 13 2 9D046796D9C4B604AE573354275A76FA6CDA851AD2DE34D61F8C51181176DAF1
```

**Zona Primaria**  
DS: Delegation  
Signer  
(Hash del KSK)

# PROPUESTA

Implementación con: BIND 9.18, Ubuntu 22.04, WEBMIN

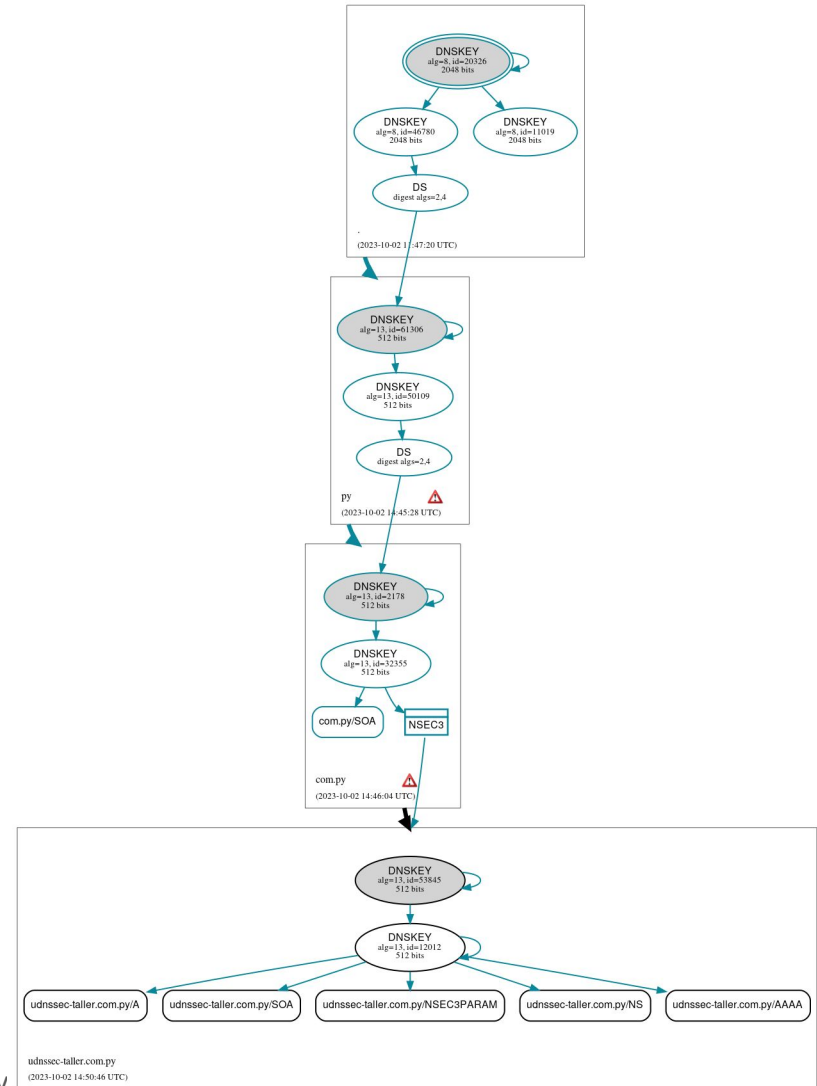
```
root@vm-taller:/var/lib/bind# pwd
/var/lib/bind
root@vm-taller:/var/lib/bind# ls -l *udnssec-taller.com.py*
-rw-r--r-- 1 root root 106 oct 2 05:02 dsset-udnssec-taller.com.py.
-rwxrwxr-x 1 root bind 362 oct 2 04:54 Kudnssec-taller.com.py.+013+12012.key
-rwxrwxr-x 1 root bind 187 oct 2 04:54 Kudnssec-taller.com.py.+013+12012.private
-rwxrwxr-x 1 root bind 361 oct 2 04:54 Kudnssec-taller.com.py.+013+53845.key
-rwxrwxr-x 1 root bind 187 oct 2 04:54 Kudnssec-taller.com.py.+013+53845.private
-rwxr-xr-x 1 root bind 3105 oct 2 05:02 udnssec-taller.com.py.hosts
root@vm-taller:/var/lib/bind#
```

# ENCADENAMIENTO

## Análisis preliminar

DNSVIZ

<https://dnsviz.net/>



# ENCADENAMIENTO

## Análisis preliminar



Back to Verisign Labs Tools

Domain Name:

Detail: [more\(+\)](#) / [less\(-\)](#)

Time: 2023-10-02 14:53:35 UTC, NTP stratum 4

Analyzing DNSSEC problems for [dnssec-taller.com.py](#)

DS=20326SHA-256 is now in the chain-of-trust	
Checking DS between Trust Anchor and	
. IN DS ( 20326 8 2 e86d44b80b8f1d39a95c0b0d7c55d08458e880409bbc683457104237c7f8ec8d )	
Found 3 DNSKEY records for:	
. 172800 IN DNSKEY ( 256 3 8 AwEAAcEtWAt103exfN3A+wnZL7AVEBA3cRrrwKTFnmcdLT9C3DC0F1Xz54s84v90xv10Vg+T R8VpkuYynMpb3POXLomLxe7j7P1xqEBo1LQ0aWfayymCuF7KebTLB5yZcR8/QU0Pwk/weLgyuX61VN D jI20meMqPKGjtDFp+EUfK90a8AuLKPLtSUL1W+IFT8sqro5XKXvXhKEZNI01z8XTG/z9xjwLpC d3/LSPUpEC9+rU2X1uCq8mAwJ7w6Rc49R XU095vc57HKqHj NuapvG0PCYh3bXCQz2b7mbZpNwu eRMFTzP51EzXWZD71zLg9j3veJ21a/WIEtS/Tp9GI0M= ) ; Key ID = 11019	
. 172800 IN DNSKEY ( 256 3 8 AwEAAAdd595Rv5uItKUCN7vypb8kDZgmtXwN55j/d08+X7ND2sgWbaBKnFhftr0s5X9DUhKR3gpM PIxAc84Nou8Wzkiu2A/sTzP1F6KpCL8epgemd1Zv41ATHEjpb0KHIOmDjSE0/frGg181j02v0F3A MSrUwH7qntL1E5uPHGKRm+agqghcAYfJHJ1dW7K13Fo2R083VZbXU9yJ3vL/T4hngel7zK84v g162tL1Jw1rK55/3U4p/bZarjTMFOHDfh0DEj1ywtRpkpPnge03gmInoa2tz+Kf767kbQb0NhHJY zPRpV1aMEWZ19pggH92yufdNfRrx68XS107sya7/1+<= ) ; Key ID = 46780	
. 172800 IN DNSKEY ( 257 3 8 AwEAAaz/tAm8yTn4Mfeh5eyI96W5exTBavKmgJzkKTO1W1vkTbxzef3+/4RgWdQ7HrxR1xH1FLE x0LAJr5emLwN75WxgnL4+B5x0LWvz80gBkVAfMtNR0xV0uCaSnId0d5LKyWbRdZn9Wge2R8Pzgc mR3EqVLrjyBxWezF0jLHwVN8efS3rCj/Ewgv1Wgb9tarpVUDK/b580a+saqLs3eNbuV7pr+eoZG+ SrdK6nW6L3c6H5Apzx7L1V1cUIdS1Xxu0LYA4/118mSVIzuDmfdrUfhHdY6+cn8HFRm+2hM8AnX Gxws9555KrUB5q1hy1Ga8subXZ2N6UwNR1AKUTV74bl= ) ; Key ID = 20326	
DNSKEY=20326SEP is now in the chain-of-trust	
DS=20326SHA-256 verifies DNSKEY=20326SEP	
Found 1 RRSIGs over DNSKEY RRset	
. 172800 IN RRSIG ( DNSKEY 8 0 172800 20231022000000 20231001000000 20326 . W0+tzT8gKOYmaX8NLjusFmLz4zVeeY3hAmZUKx17/6LxY31jh6qRiGLNB7EKAhYtjVvryUeY ExIX/LuU4fGmUCyxhYzWf+2bcv3uT4DaI8ssk3USVFFOR9wTkkgzVYrtr2UH1PFm05e//a1Pfsy NEA1sEXMZ3nUT/aCd5o26L1H53Nz9C20a0vNHTu6erIq+mhA0zshcG9/B1fCrYJ1H505serTV1x t3Tvz2+KX6w3dG1Nb6vNpe9YJ7Eb0885f1qzoAM5jcwH1Y+rovSbdVpmD3mIHxZ65vc1B6a6v05 nJ0XwchjWQpN5DFGuZhd3d189C65erv9grHteQ= )	
RRSIG=20326 and DNSKEY=20326SEP verifies the DNSKEY RRset	
DNSKEY=11019 is now in the chain-of-trust	
DNSKEY=46780 is now in the chain-of-trust	
Found child zone py	

## DNSSEC Analyzer

<https://dnssec-analyzer.verisignlabs.com/>

# ENCADENAMIENTO

## Declaración de Registro DS - NIC.PY

### Modificación de datos de dominios vía web

Nombre de Dominio: udnssec-taller.com.py

#### Observación:

- El contacto solicitante de la modificación es el **Contacto Administrativo**. Por lo tanto, está habilitado a modificar sólo los datos que se encuentran en éste formulario.
- En este formulario sólo se podrá registrar y/o actualizar los registros DS asociados al dominio. La información de servidores DNS y contactos se muestra con fines informativos.

#### Registros DNSSEC

##### Sección DNSSEC - Registro DS

##### Registro DS 1

Etiqueta de la llave DS

(\*) Valor entero menor que 65.536 de identificación del registro DS

Tipo de algoritmo criptográfico del DNSKEY

8-RSA/SHA-256  13-ECDSA/SHA-256

Resumen de la llave DS (digest)

```
9D046796D9C4B604AE573354275A76FA6CDA851AD2DE34D61F8C51181176DAF1
```

Dar de baja registro DS

Agregar nuevo registro DS

udnssec-taller.com.py. IN DS 53845 13 2  
9D046796D9C4B604AE573354275A76FA6CDA851AD2DE34D61F8C51181176DAF1

**NIC.PY**

<https://www.nic.py/modificaciones/>

# ¿Iniciamos el proceso con tu dominio?

✓ Escribenos, y preparamos un plan de trabajo...



[dnssec@nic.py](mailto:dnssec@nic.py)





Universidad Nacional de Asunción  
**CNC**  
Centro Nacional de Computación

**NIC.py**

**32** AÑOS  
1991-2023  
**.PY** te conecta  
al mundo

Universidad  
Católica  
*"Nuestra Señora de la Asunción"*

**LED**  
LABORATORIO  
de Electrónica  
Digital



`dnssec@nic.py`